Proceedings in Informatics and Information Technologies

Student Research Conference 2013

Mária Bieliková (Ed.)

Keynote by A Min Tjoa

STU FIIT

Proceedings in
Informatics and Information Technologies

IIT.SRC 2013
Student Research Conference

Mária Bieliková (Ed.)

IIT.SRC 2013: Student Research Conference

9th Student Research Conference
in Informatics and Information Technologies
Bratislava, April 23, 2013

Post-Conference Proceedings

IEEE Czechoslovakia Section
acm Slovakia Chapter

SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA
Faculty of Informatics and Information Technologies

Proceedings in
Informatics and Information Technologies

IIT.SRC 2013
Student Research Conference
Post-Conference Proceedings

Editor

Mária Bieliková
Faculty of Informatics and Information Technologies
Slovak University of Technology in Bratislava
Ilkovičova 2
842 16 Bratislava, Slovakia

Visit IIT.SRC on the Web: http://iit-src.stuba.sk

Executive Editors: Katarína Mršková, Anton Andrejko
Copy Editors: Andrej Fogelton, Dávid Chalupa, Peter Jombík,
Štefan Krištofík, Tomáš Kučečka, Martin Labaj, Dominik Macko,
Michal Olšovský, Márius Šajgalík, Martin Vojtko
FIIT STU
Cover Designer: Peter Kaminský

Published by:
Nakladateľstvo STU
Vazovova 5, Bratislava, Slovakia

ISBN 978-80-227-4111-8

Preface

This volume contains the keynote and comprehensive information on student papers selected for presentation and presented at IIT.SRC 2013, the 9th Student Research Conference in Informatics and Information Technologies, held April 23, 2013 at the Faculty of Informatics and Information Technologies of the Slovak University of Technology in Bratislava.

We included in this volume abstracts of all 58 full papers presented at the conference, 35 of which are included also in their full version, 19 extended abstracts, and information on accompanying events. 6 full papers were already published in the Special Section on Student Research in Informatics and Information Technologies of the Information Sciences and Technologies in the Bulletin of the ACM Slovakia (Vol. 5, No. 2, 2013, slovakia.acm.org/bulletin/). Authors of 23 full papers politely declined our invitation to publish their paper in this volume as they have already acceptance or have submitted their papers to peer reviewed scientific journals or proceedings of mostly international scientific conferences. We included together with the abstract of each paper the information on presentation elsewhere status available at time of publication of this volume. There are even cases of other authors who were able to write new papers based on a substantial expansion of their papers submitted to our student research conference and submit them elsewhere. Some of them have got their new paper already accepted for publication at time of publication of this volume.

Research has been one of the main priorities of the university education since its very beginning. It is the case also for our university – the Slovak University of Technology in Bratislava and its faculty – the Faculty of Informatics and Information Technologies. Close connection of research and education leads very naturally to a participation of students in research. This holds not only the students of doctoral study, where research is a substantial part of their study and one of their principal activities. A participation of students in research is "going down" to students of master, even bachelor study.

Universities of technology have a long tradition of students participating in a skilled labour where they have to apply their theoretical knowledge. The best of these results were usually presented at various students' competitions or exhibitions. There were also combined with student research works. Our university has a long tradition in such competition named ŠVOČ (abbreviation of the Student Scientific and Technical Activity). Nine years ago our faculty, FIIT STU, decided to transform former ŠVOČ into the Student Research Conference covering topics of Informatics and Information Technologies (IIT.SRC). Participants are students of all three levels of the study – bachelor (Bc.), master (Ing.) and doctoral (PhD.) study. The conference adopted a form of reviewing as at any other scientific conference, and presenting internally the papers in a form of internal Proceedings, which in most cases means a first step towards later publishing the results on national or international established conferences or journals.

IIT.SRC 2013 attracted 85 university student papers from which 79 were accepted (12 bachelor, 32 master, 33 doctoral). The number of papers slightly varies each year. This year we have noticed little decrease in bachelor and master categories and an increase in doctoral category comparing to IIT.SRC 2012.

IIT.SRC 2013 was organized in five sections with papers in two categories – full papers and extended abstracts:

- Intelligent Information Processing,
- Web Science and Engineering,

- Computer Graphics and Computer Vision,
- Software Engineering,
- Computer Networks, Computer Systems and Security.

The conference was opened by A Min Tjoa followed by a keynote titled *The use of Open Linked Data for decision making*. A Min Tjoa is currently full professor at the Vienna University of Technology and Director of the Institute for Software Technology and Interactive Systems. His main research interests are in databases, data warehouses, semantic web, IT security and software engineering.

Besides the 77 papers presented at the conference and included in these Proceedings several accompanying events were organized. The RoboCup Exhibition is organised as a part of IIT.SRC from 2005. RoboCup is an attractive project with free participation, designed to support education and research in artificial intelligence, robotics and information technologies. Through several years, our students achieved interesting results, which were presented during the conference. RoboCup exhibition presented both the way the RoboCup simulated league is played and also the progress of current students' research in this field. Four years ago a new RoboCup league – three-dimensional (3D) robotic simulation was added. The extension of the simulation to the third dimension shows the continuous progress in RoboCup and in our students' skills.

This year we organized for the fifth time as part IIT.SRC a showcase of TP-Cup projects. TP-Cup is a competition of master students' teams aimed at excellence in development information technologies solutions within two semester long team project module. The competition has four stages. 11 teams managed to achieve this stage and presented their projects during the TP-Cup showcase. Extended abstracts of their projects are included in these proceedings.

Accompanying events included for sixth time also our programming contest. It follows a long tradition at the Slovak University of Technology in Bratislava and our faculty in organizing programming contests, especially the ACM International Collegiate Programming Contest like competitions. This year we have organized for the second time the final round of the ProFIIT programming contest for high school students in parallel with IIT.SRC. Our aim was to show our potential future students exciting research opportunities awaiting them at our university.

We continued this year with FIITApixel exhibition. FIITApixel brings together both students and staff of the Faculty as well as its potential students and alumni in an effort to create, share and judge pictures. It is organized as an ongoing event, where anyone can contribute pictures. The IIT.SRC FIITApixel exhibition presented the best pictures of this year contest.

For second time we organized this year Junior IIT.SRC. It provides a room for presenting inventive high school student projects within the topics of the conference. Six high school students' submissions were selected, what in quantity doubles the pioneer track from last year. Two of these projects are also presented as extended abstracts for more detailed explanation of proposed ideas and realized prototypes in this volume.

IIT.SRC 2013 was for the first time organized in new FIIT building. We all benefited from well-disposed space, which supported live discussions. IIT.SRC 2013 is the result of considerable effort by a number of people. It is our pleasure to express our thanks to:

- the IIT.SRC 2013 Programme Committee who devoted effort to reviewing papers and awards selection,
- the IIT.SRC 2013 Organising Committee and accompanying events coordinators (mentioned in particular reports in these proceedings) for a smooth preparation of the event,
- the students – authors of the papers, for contributing good papers reporting their research and their supervisors for bringing the students to research community.

Special thanks go to:

- Katarína Mršková together with Anton Andrejko who did an excellent job in the completion of the proceedings,
- Zuzana Marušincová and the whole organizing committee for effective support of all activities and in making the conference happen.

Finally we highly appreciate the financial support of our sponsors which helped the organizers to provide excellent environment for presentation of the results of student research and valuable awards.

Bratislava, November 2013

Pavel Čičák and Mária Bieliková

Conference Organisation

The 9th Student Research Conference in Informatics and Information Technologies (IIT.SRC), held on April 23, 2013 in Bratislava, was organised by the Slovak University of Technology (and, in particular, its Faculty of Informatics and Information Technologies) in Bratislava.

IIT.SRC General Chair

Pavel Čičák (dean, Faculty of Informatics and Information Technologies,
Slovak University of Technology in Bratislava)

IIT.SRC Programme Chair

Mária Bieliková

IIT.SRC Programme Committee

Michal Barla
Vanda Benešová
Anna Bou Ezzeddine
Michal Čerňanský
Pavel Čičák
Peter Drahoš
Jana Flochová
Elena Gramatová
Ladislav Hudec
Daniela Chudá
Katarína Jelemenská
Peter Kapec
Gabriela Kosková
Margaréta Kotočová
Ivan Kotuliak
Tomáš Kováčik
Alena Kovárová
Tibor Krajčovič
Vladimír Kvasnička
Peter Lacko
Michal Laclavík
Ján Lang
Marián Lekavý
Mária Lucká
Pavol Mederly
Ľudovít Molnár
Pavol Návrat
Mária Pohronská
Ivan Polášek
Jiří Pospíchal
Viera Rozinajová
Petr Šaloun
Marián Šimko
Juraj Štefanovič
Jozef Tvarožek
Valentino Vranić

IIT.SRC Organising Committee

Alexandra Bieleková
Mária Bieliková
Ivan Kotuliak
Zuzana Marušincová, *Chair*
Michal Michel
Katarína Mršková
Ľubica Palatinusová
Branislav Steinmüller
Roman Stovíček

all from FIIT STU in Bratislava, Slovakia

IIT.SRC General Sponsors

- Capco Slovakia
- Hewlett-Packard Slovakia
- IBM Slovakia

Supporting Professional Societies and Foundations

- ACM Slovakia Chapter
- Czechoslovakia Section of IEEE
- Slovak Society for Computer Science
- Informatics Development Foundation at FIIT STU

Medial Partner

- PC Revue

Table of Contents

Keynote

Intelligent Information Processing, Volume 1

Bachelor Degree Program Students

Master Degree Program Students

Doctoral Degree Program Students

Web Science and Engineering, Volume 1

Computer Graphics and Computer Vision, Volume 1

Software Engineering, Volume 2

Computer Networks, Computer Systems and Security, Volume 2

Extended Abstracts, Volume 2

Accompanying Events, Volume 2

The Use of Open Linked Data for Decision Making

A Min TJOA

Institute of Software Technology and Interactive Systems
Vienna University of Technology
Favoritenstrasse 9-11, 1040 Vienna, Austria
amin@ifs.tuwien.ac.at

Abstract. The scattered information on the Web can form a global data graph that connects distributed resources and facilitates the discovery of new resources. In this context "Linked Data" introduces some simple and effective principles for publishing and connecting structured data on the Web. "Linked Data" has gained momentum among governments, in the academic and business world, and in the public sector over the last few years. Today a growing number of high quality and public "Linked Data" resources are published on the Web which can benefit the decision makers and authorities at the national and international levels to overcome the data gaps and improve the information availability. In this context, the recent advancements of "Linked Data" and "Linked Open Data" (LOD) approaches for capturing, managing, and distribution of information will be explored and their potential for addressing the decision makers' requirements will be highlighted.

1 Introduction

Machine-readable datasets which are liberated from the proprietary tools and formats are the key enablers for building innovative application and services. Linked Data as a set of best practices for publishing and connecting structured data on the Web, is aiming to evolve the current Web of documents into Web of Data. In this context, semantic technologies have established a solid basis for Linked Data by setting up the semantic languages and standards. During the past few years, the Linked Data initiative has been constantly growing and credited by the scientific community, governments and policy makers. A result of these activities is the huge amount of high quality linked datasets that are made available to public following the Linked Data principles for identifying, publishing, linking, and discovery of date entities over the Web.

2 Governmental Linked Data Initiatives

There are a number of governmental initiatives that are trying to keep pace with the changes in Linked Data technologies and follow the interoperability and openness principles. Such initiatives follow a fundamental shift from traditional information management models to a new shared data-centric model where instead of managing documents, governments will focus on managing discrete pieces of open data and content. Open Data in the governmental sector, also known as Open Government Data (OGD), aims to establish a modern cooperation among politicians, public

administration, industry, and private citizens by enabling more transparency, democracy, participation, and collaboration.

3 Linked Data Challenges

Although the Linked Data concept has opened up lots of possibilities for data sharing and data collaboration, the data integration process based on the existing raw data is still a challenging task that needs an in-depth insight of underlying datasets together with high technical expertise. A number of LOD consumption challenges that are hampering the effective consumption of Linked Data, are briefly listed in this section.

3.1 Schema Heterogeneity and Lack of Schema-level Links

Due to the open and distributed nature of LOD, the datasets do not need to follow a uniform conceptual schema and may define their own concepts locally. As a result, a single real-world concept might have multiple representations in different LOD datasets, which in most of cases are not connected via appropriate schema-level links. Therefore LOD solution providers need to explore the precise details of the target LOD datasets and enrich the datasets through nontrivial cross-links in order to create their unique data integration solutions. Unfortunately, the creation of user-generated solutions demands considerable effort in familiarizing oneself with the target datasets, and due to the goal-oriented nature of these data integration solutions, the results cannot be efficiently shared and reused.

3.2 Lack of Custom-tailored LOD Tools

Current LOD tools and frameworks typically address the generic requirements of Linked Data solutions such as LOD publishing, storage, query, and reconciliation. It can be predicted that in most software (application) domains that the one-size-fits-all era has come to an end and the LOD consuming challenges are indicative for the end of this generic attempt. The LOD community is now striving for custom-tailored tools for addressing the long tail of requirements needed by a large number of novice end-users and solution providers.

3.3 Effective LOD integration with Open APIs

The Open API is an important recent trend in social media and Web 2.0. Many service providers incorporate Open APIs to offer their data and core functions for data integration and lightweight service composition use cases. Today, there is a growing number of Open APIs that, like LOD, provide well-structured data in a scalable and resource-oriented manner. Unfortunately, neither Linked Data nor the Open APIs sufficiently describe the integration of these two information spaces. The LOD community has tried to overcome this problem by introducing Linked APIs. However, the data integration environments need to be improved to take the full advantage of both LOD and Open APIs for creating effective data integration solutions.

4 Conclusions

The applications that take benefit of Linked Data are not yet as widespread as expected by its initiator Sir Tim Berners-Lee. One of the most important causes for this situation is the still relatively high necessary workload for data gathering, processing, and integration in context of Linked Data for the solution providing and stakeholders such as governments and enterprises. The current state-of-the-art shows that Linked Data consumption is still an open issue. Lowering such entrance barriers is therefore imperative for evolution and development of Linked Data. Research and development have to focus on lowering the burden which is still necessary to provide the desired interoperability.

Intelligent Information Processing

Improving Speech Therapy by Motivational Home Exercises

Peter DEMČÁK, Ondrej GALBAVÝ,
Miroslav ŠIMEK, Veronika ŠTRBÁKOVÁ*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
Icup2013@googlegroups.com

Abstract. This paper discusses our solution aimed at improving speech abilities of children by enriching speech therapy with tools supporting individual speech exercises. These tools consist of a unified platform which provides communication between therapist and patient and motivational means for the patient in the form of serious games. This paper further analyzes the ways in which this motivation is achieved, as well as types of different speech therapy exercises, which the serious games in our solution emulate. Then, we will touch the topics of computer vision and sound analysis that we use to enhance our games.

1 Introduction

The ability of speech has an important place in human lives. It is closely connected to other human skills, notably thinking, perception, kinetics, feelings, learning, writing and reading [1].

Nowadays, many people and especially children suffer from speech disorders. Among the most known forms of speech disorders we can name stuttering, cluttering or incorrect pronunciation. These disorders have a negative effect on the quality individual's life. It is essential that children perfect their ability to speak correctly before they begin attending primary school, because prolonged speech disability has a long lasting impact on the child's mentality, social life and future achievements.

At school, children with speech disabilities are easy targets for bullying, which exposes them to higher level of stress, and can lead to development of shyness and distrust towards others. Some children, who are affected by speech disorders based on deficient hearing differentiation generally confuse individual letters and have difficulties with writing words correctly.

The most obvious way to improve child's speech ability is by attending a speech therapy. Although this therapy may be in many cases effective, the statistics still show that about 40% children in our primary schools have some kind of speech disorder, which persists in 10-15% cases until secondary schools [2]. The cause of these persistent problems comes from the fact, that children visiting speech therapists must practice exercises at home while these exercises are

* Bachelor degree study programme in field: Informatics
Supervisor: Dr. Michal Barla, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 1–6.

incredibly boring. This means that children cannot keep their attention for longer periods of time and are easily interrupted by any other surroundings factors. Even if parents force their children to practice, they often capitulate too quickly, resulting in an unsuccessful speech therapy.

2 Related work

Since speech disorders are well known, many innovative solutions have been created for the purpose to improve speech abilities of children by using them. These solutions generally take two different approaches:

1. Speech therapy enhancement systems – These are software products used by speech therapists in their offices (e.g., FONO, PILP). They provide various tools to help the therapist improve their work. However, they are often complicated, accessible only to professionals, unusable to children without the help of therapist. Thus, they are practically bound by the same limits as normal speech therapy: limited time per patient and place of use restricted to the therapist's office
2. Speech exercise applications – There are several applications on the market, notably games (e.g., Say-N-Play[1], IcSpeechGames[2]), proclaiming their beneficial effects on children's speech capabilities. Regardless of their benefits however, it is good not to forget that speech therapy is a complex process which in serious cases requires a real specialist who – as of the time of writing this paper – cannot be replaced by software. What these sorts of applications realistically lack is a way to communicate child's progress with a real speech therapist.

3 Concept overview

In order to improve efficiency of speech therapy, we devised Speekle, a system focused on support of home exercises and their integration into the whole therapy process. Speekle consists of a client application *TalkLand* which supports individual speech exercises in form of games and a unifying server application, which provides communication between a speech therapist and his or her patients.

From child's point of view, Speekle turns boring exercises into specially conceived serious games, which include proper motivational factors. Games are focused on different kinds of exercises like pronunciation, speech sound differentiation and training of tongue muscles, which are some of the most common types of exercises used to achieve progress in speech therapy.

Speekle provides benefits for speech therapists as well. As a child practices in Talkland, a speech therapist gets statistics on performed exercises including recordings of key moments of the exercise. This way, the speech therapists have a complete overview of child's efforts and feedback on efficiency of the therapy and thus can prepare themselves for the upcoming appointment.

One advantage of our solution, compared to related works is that children can do exercises at home without presence of speech therapist as they get an immediate in-game play feedback. In order to do this, we must be able to capture and evaluate child's effort – pronounced sounds and tongue movements.

4 Sound analyzer

There are several approaches to analyze sound, the one we use in the Speekle sound analyzer is based on frequency decomposition of sound signal. Every waveform is a combination of sine

[1] Say-N-Play, http://www.saynplay.com/
[2] IcSpeechGames, http://www.rose-medical.com/speech-therapy-games.html

waveforms with different frequencies and different amplitudes over time. This frequency decomposition can be obtained with Fast Fourier Transformation (FFT) algorithm [3].

The current version of our Speekle sound analyzer recognizes continuously pronounced sibilants. The sound of sibilants consists mainly of noise. A noise does not have any specific dominant frequencies in contrast to a tone, which consists of harmonic frequencies. All harmonic frequencies are integer multiplications of a fundamental harmonic frequency which is the lowest dominant frequency in tone. However, a noise contains dominant frequency bands, which we take advantage of. Speekle sound analyzer recognizes a signal as a correctly pronounced sibilant if it contains frequencies in the dominant frequency band of the sibilant that is being recognized. The signal also must not contain frequencies outside of this dominant frequency band.

There are two main types of sibilants for Slovak language. Voiced (z, ž) and voiceless (s, š). Voiced sibilants differ from their voiceless counterparts in the fact that they also contain harmonic frequencies. Therefore, even though they still consist mainly from noise, voiced sibilants carry tone characteristics as well. This implies the necessity, for checking whether the signal contains harmonic frequencies or not during their recognition.

The output of FFT is a very rough curve, so first of all it needs to be smoothed. Speekle algorithm for smoothing computes values in each point as an average of N values around. The N must be high enough to provide enough smoothness but it cannot be too high to keep enough details to the curve. This is necessary especially for finding harmonic frequencies because they are very close to each other and with too high N the peaks of harmonic frequencies will get lost.

With smooth curve it is easier to find its significant local extremes. Local extremes are very useful for finding harmonic frequencies and also for finding dominant frequency bands. However, finding all the local extremes would be practically useless, because even though the curve is now much smoother, it is still far from being entirely smooth.

Our algorithm for finding only the most significant local extremes works by finding the extremes by passing the array from left to right remembering the extreme value (the highest or the lowest one based on current state of the algorithm whether it's searching for local minimum or local maximum) along with its index. When the difference between the extreme values and value in current position crosses certain limit (this limit is an average value of whole curve or the certain fraction of the maximal value), then the highest or lowest value is stored as local extreme. Value and index of this extreme is reset and the searching state of algorithm is changed to the other extreme. Finding the first extreme is a special case when we do not yet know, for what type of extreme we are looking for, so the algorithm is searching for both the maximum and the minimum extreme.

Since the sound signal is analyzed in real time, we analyze it in chunks. Each chunk is 4096 samples wide which based on 44100 sample rate means it's approximately 92 milliseconds long. This length of chunk provides sufficient response time and enough data to work with. Analysis consists of these two separate parts: harmonic analysis and noise analysis.

The result of harmonic analysis is whether the current chunk does or does not contain harmonic frequencies. Because harmonic frequencies are very significant peaks, the easiest way to recognize them is from the significant local extremes specifically local maximums. However, local maximum alone may not be at the center of the peak, so first of all it is important to find these correct central frequencies of peaks. This is done by going to the left and to the right from the local maximum until the value of the curve reaches a certain fraction of the local maximum. The frequency of harmonic is the average of these left and right peak boundaries.

To prevent the distortion caused by noise presence over harmonics, local maximum is considered to be harmonic only in range from 100 Hz to 1500 Hz. In correctly pronounced sibilants there shouldn't be much noise in this range so the harmonics should be clearly distinguishable. The lower boundary of 100 Hz is there to eliminate possible incorrect harmonic frequencies detection caused by blowing into the microphone.

After we have processed all local maximums into possible harmonics, we assess whether the possible harmonics are real harmonics which have to be integral multiplications of the first

harmonic in terms of their frequency. Of course there needs to be right amount of approximation, so in our algorithm the harmonics are considered to be integral multiplication of the first harmonic only as long as the decimal part of the ratio is in the interval <0.0,0.25> ∪ <0.75,1.0).

One more approximation is done on harmonics. They all may be shifted for a certain same small value which may be positive or negative. This shift value is computed by following equation:

$$\text{frequencyShift} = \max(-\text{limit}, \min(\text{limit}, \text{secondHarm} - 2 * \text{firstHarm}))$$

Our analysis of noise is a combination of many different approaches. The main one consists of finding the dominant frequency band. More accurately, whether the dominant frequency band is located where it should be. The way dominant frequency band is being found differs from sibilant to sibilant but basically the idea is to find certain local maximum, for example the absolute maximum. From there, we expand the band to both sides until the value is lower than a certain limit which is the square root of the absolute maximum of the curve. This algorithm returns a result of four attributes that are being used for deciding whether given chunk is, or isn't correctly pronounced. These attributes are namely: the size of the band and the minimum, maximum and middle frequencies of the band.

For Speekle, the real time aspect of the sound analysis is very important, because it enables us to provide instant feedback to children. Moreover, Speekle also provides feedback to overseeing speech therapists. Basic data that we pass on to the speech therapist are success rates and duration of game sessions.

Furthermore, speech therapists prefer having more direct knowledge about the happening during the exercising process. Passing whole recordings wouldn't be practical for two reasons. Firstly the storage space costs would be too high over time, and secondly it would produce overloads of useless material, since speech therapists do not need and want to listen through hours of unfiltered recordings.

The solution we created for this problem is called key moments. Key moments are samples that contain selection of potentially interesting moments for speech therapist. These moments are the best and the worst moments of whole session and a selection of bad but partially good moments. This filtering is done based on success rates of specific key moments. First of all, potential key moment must be long enough and have high or low enough success rate to be even considered. If it satisfies these conditions it is compared with other key moments currently considered for final set of key moments and if it is better in some category, it will replace the worst key moment in that category. The max amount of key moments is limited by logarithm of current length of session.

Our development of sound analyzer was based on many different samples containing correctly as well as incorrectly pronounced sibilants from both children and adults. We developed a special tool which helped us to build training and testing sets by manually annotating mentioned samples. It displays each sample as an image of spectrogram which is divided into chunks and each chunk is colored the way that means how the chunk would be analyzed in that particular moment by real time analysis. A wide set of samples was annotated by manually passing through each one of them and deciding whether the sample is correctly or incorrectly pronounced and what letter is actually pronounced on given sample. This manual annotation was required as there is no better way to distinguish how the sample sounds to human ear than by actual human ear. Thanks to annotated set of samples it was possible to automatically test the samples and receive images and names of currently problematic samples that do not match with their annotation.

5 Tongue tracking

We have considered two ways to implement tongue tracking. The first we have worked on involved video feed acquisition using standard computer web camera. Video feed of resolution

640x480 pixels is processed using open source computer vision library OpenCV. We use Haar-like cascade filters to recognize face patterns in each video frame, so we identify present faces.

Next we specify where mouth could be located by constants, which does not have to be precise. Reason, why we are reducing mouth search area is that it is not effective to search mouth all over the face with a time-consuming algorithm. Then we can search for mouth patterns using Haar-like filters. Situation gets complicated when we are tracking tongue. We have not implemented a working algorithm of this type yet.

We have considered color separation to separate lips and tongue from skin and teeth and following surface separation. Then we would have individual surfaces with their positions and areas. With this information we could determine which of these surfaces is representing tongue and determine its position from center of mouth. Another possible option is to train multiple Haar-like filters with images of tongue moved left, right, up, down, centered and closed mouth. Then we could search for these patterns and determine several tongue binary parameters (left, right, up, down, sticked out). We will look further into these approaches later. Another problem worth mention is various qualities of low cost web cameras sold on mass market. Algorithm can be sensitive to video feed quality, resolution, frame rate and tracked face illumination.

The second approach we have worked one used Microsoft Kinect for data acquisition. Kinect captures a depth map in addition to a color frame. Depth map has resolution 640x480 with 11-bit depth information. Kinect can reliably sense depth from 0.7 meters with 1.3mm precision, which is usable for tongue tracking. Depth map is obtained by projecting special patterns in infrared spectrum and capturing them by infrared camera placed in certain distance from projector. Software then computes depth map from distortions of patterns. Microsoft also provides Face Tracking SDK, which computes additional variables to detected faces, like 3D head pose, facial gestures parameters and set of 108 points on face (upper lip top, outline of eyes etc.) and coordinates of them in color video frame.

We use these additional data in tongue tracking. At first, we map depth map to a color frame coordinates, as they are created with two separate alongside cameras with different field of view. Subsequently, the tongue tracker locates closest point in depth map to Kinect sensor in mouth area and near it.

Once having position of tongue in color frame coordinate system, we can convert it to relative position from center of mouth. But we must correct it using 3D head pose information, because when the head is turned slightly to the left and tongue is centered, it may appear that tongue is left to the center. Our current implementation's problem is, that when tongue is lowered, it is on the same depth as upper lip which causes problems in interpreting depth data.

We have tested this algorithm manually with connected Kinect and moving with tongue in front of it. Also we can replay recordings of color and video feed via Kinect Studio, which taps into connection between application and sensor. We are working on more automated tests, which involves manually annotated recordings (values as head present, tongue sticked out and assumed tongue position) and measuring deviation of computed and annotated values.

6 Conclusions

In this paper we proposed Speekle, our solution intended for improvement of speech therapy via support of individual exercises for child patients with speech disabilities. The exercises we incorporate can be divided according to the nature of speech skill they aim to improve: speech sound differentiation, phoneme pronunciation and oral motor skills.

Our solution is based on two key principles:

- Motivation. The target demographics a.k.a. the patients with speech disorders consist mainly of children in preschool and early school age. With these children, parents and therapists find it difficult to focus their attention and keep them interested in practicing. We created the

Talkland game client application, which makes of use of colorful a child-friendly environment.

- Progress tracking. Our solution does not aim to replace classic speech therapy practices, but to enrich them and improve on interconnection between home exercising and therapy taking place in the presence of the therapist. To do this, Speekle provides therapists with tools to monitor and control home exercises of their patients.

Our game application employs real time sound analysis and computer vision. Sound analysis uses combination of different aspects like searching for harmonic frequencies and dominant frequency bands. Computer vision is using Microsoft Kinect to track tongue based on depth map.

References

[1] Vužňáková, K.: Rečová (jazyková) výchova v materskej škole a vývin detskej reči. In: *Slovo o slove*. 14. Zborník KKLV PF PU, Prešov: PF PU, (2008), pp. 150–158.

[2] Lechta, V. et al.: Teoretické východiská súčasnej logopédie, moderné prístupy k logopedickej starostlivosti o osoby s narušenou komunikačnou schopnosťou. Bratislava: SPN, (1990), 280 p.

[3] Lyons, R.: Understanding Digital Signal Processing 2. New Jersey: Pearson Education, Inc. (2004). ISBN 0-13-108989-7

Extracting Keywords from Educational Content

Jozef HARINEK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
j.harinek@gmail.com

Abstract. When considering social educational systems, we can improve results of relevant domain term acquisition from educational content by processing user created annotations assigned to the documents. The annotations provide us potentially useful information about documents and can improve the results of base Automatic Term Recognition (ATR) algorithms. We propose a method for relevant domain terms extraction based on user generated annotations processing. We consider three basic annotation types (tag, comment, highlight). We compute the final term weight by combining relevant domain terms weights obtained from the individual annotation types and those obtained from the text. We evaluated the method using data from Principles of Software Engineering course and showed that enhancements based on annotation processing yield 22.6 % improvement of results. We believe we can improve these results by taking into account even more attributes of the annotations.

A paper based in part on this paper was published in DocEng '13 Proceedings of the 2013 ACM symposium on Document engineering, ACM New York, pp. 185-188.

* Bachelor degree study programme in field: Informatics
Supervisor: Dr. Marián Šimko, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

Optimization Algorithm Inspired by Social Insect Behaviour in Comparison with Hill Climbing

Daniel SOÓS*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
soos.dano@gmail.com

Abstract. This paper experiments with an algorithm inspired by the social insect behaviour. In this method, each member of the population represents a solution to the optimization problem and the algorithm can be described as a multiagent method. Lifespans of these agents are based on the quality of the solution. One agent in every generation moves out of the nest as it seeks for food in the artificial world. Afterwards, if the case is that it found food, other agents staying in the nest will know about the food source. New solutions are generated each generation thanks to mutation. We test this algorithm in comparison with a stochastic parallel hill climbing algorithm on a typical non-convex function, the Rastrigin function and other well-known mathematic functions which we want to minimize. Our results show that the newly proposed algorithm does not work as efficient as the parallel hill climbing algorithm on the Rastrigin function, but outperforms it on the other selected function. There is also room for improvement in the presented algorithm and we suggest a new technique built into it that may work properly.

Amended version published in Information Sciences and Technologies Bulletin of the ACM Slovakia, Vol. 5, No. 2 (2013), pp. 48-52.

* Bachelor degree study programme in field: Informatics
Supervisor: David Chalupa, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 13-18.

Identification of Persons by Using Algorithms of Fuzzy Ants

Anton BALUCHA*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
a.balucha@gmail.com

Abstract. People reveal personal data, present opinions, describe experiences or show private photos. These data, once available only to a small group of people around the person are now available to a wide range of people around the world. People search information about the other people. On the process of searching information is always something to improve. We analyze possibilities of retrieving information about the person, analyze possibilities of person identification and analyze using fuzzy ants clustering algorithm for retrieving information from gained data. We present integrated information about the person and use the information for searching relevant information.

1 People on the Web

Today is the Internet most widely distributed system used for communication between people. It connects people and provides communication tools to people. People use the communication tools, such as *discussion forums* – for exchanging their views and opinions; *blogs* – for expressing their knowledge, experiences or opinions; *social networks* – for connecting with friends or with other people, *photogalleries* – to store their memories; *IM applications* – for short and immediately messages or *emails* also to communicate with other people. We can basically say that if the person actively uses these tools, we can get complex view about his personality.

By common look, we can see and find on many pages information about the same person. We can read it from the content of the page in which is searched person mentioned. But also we can see and understand that on different pages are just namesakes who have nothing in common with the searched person.

Information about the searched person is served to us by search engines in a scattered way. We have to pass along many pages to get summarized view about the person. Many times is the same information written by different ways or in different data formats. Although information may be written by many ways, it is possible to read the scattered information, joint it and get it into integrated form. We analyze possibilities of person identification analyze using fuzzy ants clustering algorithm for retrieving information from gained data. We present integrated information about the person and use the integrated information for searching information.

---

* Master degree study programme in field: Information Systems
Supervisor: Dr. Anna Bou Ezzeddine, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 19–24.

2 Personal data on the Web

Every person brings with himself many various personal data through which is possible to identify him. We identified elementary personal data used for identification of person such as *name* (first name, surname, initials); *titles* (before name, after name); *addresses* (permanent address, temporary address); *personal identification number*; *number of identity card*; *number of passport; identification numbers of companies; telephone numbers* (private or business); *e-mail* (private or business e-mail); *nick*; *OpenID* and *URL* (personal pages or blog). For every one of the elementary personal data we created a regular expression which helps us to retrieve personal data from text or web page.

After identification of personal data we analyzed following properties of the personal data: *name of personal data; description of personal data; usage, possible application of personal data in text; way of identification personal data; metainformation* (information about everything else – type of text, type of web page, etc.). These properties can help to better understand of usage of personal data.

This personal data can very precisely identify person on different web pages. Although content of web pages may not be similar and identification based on keywords may fail, identification based on personal data can bring better search results because of the nature of personal data – many times they are unique, they belong to one person and serve to identification. It is also possible to uniquely determine them by regular expressions, which we also apply on them.

2.1 Social networks

Another way of getting personal is using social networks. Many of them provides API through which is possible to retrieve various personal data. We identified around 100 social networks but we focused only on two biggest social networks which provide API for getting personal data.

2.1.1 Facebook

Facebook provides on *https://developers.facebook.com/* very detail description of possibilities how to get and manage personal data through its five different API – Login, Open Graph, Graph API, FGL and Rest API. It describes how to get information and manipulate with personal data, photos, video etc. Due to its simplicity we incorporated it into our project to help search people.

2.1.2 Google+

API for Google+ is available at *https://developers.google.com/+/api/*. It provides much more information about person in a much cleaner API than the Facebook API. Because it is based on REST web services and because of its simplicity we also incorporated it into our project to help search information about people.

2.1.3 Other social networks

Because of huge number of social networks we discovered, it is not possible to concentrate on every each of them and study their API. This is the reason why we use regular expressions to find elementary personal data on the social networks and on the other web pages.

3 Existing solutions

We identified many applications which help us to search persons. But many times are search results created by the applications relevant just for certain country, search engines do not have world-wide coverage or they are just a desktop application. After making research of available search engines we defined basic features of our application. So we create a web application

available for every user which it uses more sources for searching of persons such as social portals and uses clustering algorithms for identifying of person.

4 Pre-processing of searched text

Before we apply algorithms of clustering, it is necessary to pre-process searched or processing text and evaluate web documents. We process and evaluate these features:

- *identification of keywords and its number* – if two documents has same keywords, we can say that the measure of equality is high and we can group it together
- *identification of web server* – if two documents are located on the same web server it is highly possible, that they belongs to same person
- *identification of links to other documents* – if documents are linked to same pages, it is also possible, that in the documents is mentioned same person
- *metainformation gained from HTML tags* – HTML tags bring us various metainformation about documents, which help us to better understand the content of web page.

For getting the keywords from documents we created custom library which helps us in this process. In enables removing unnecessary HTML tags, removes stop words, it put words into base form for which help us also custom stemmer. For indexing of words and making statistics used for text evaluating we use Apache Lucene. Than the library counts the TF-IDF statistics for each word and identify keywords.

5 Process of clustering

During the work on the project we created two applications, which demonstrate text processing and evaluation of document in *searchPerson* and clustering by fuzzy ants algorithm in *clusteringAnts*.

5.1 Clustering of documents using searchPerson application

searchPerson is web application available at *http://www.tonyb.sk/search.jsp*, which uses results of other search engines to get data about persons which are subsequently processed. Text of search results are initially processed as was mentioned in the chapter *Pre-processing of searched text.* Than we create a matrix with *NxN* dimensions where *N* is a number of searched documents. Into the matrix we notice the results of document evaluation. Evaluation of two documents is based on similarity of keywords. The more keywords are similar, the better evaluation gets the couple of documents. Couples which get highest evaluation are pages on which is the same person. Result of this process is group of clusters, which contains web pages on which we assume same person.

5.2 Clustering by fuzzy ants using clusteringAnts application

clusteringAnts is a standard desktop application, which uses fuzzy ants algorithm for clustering needles. It uses defined algorithm and several input parameters which modify its behavior. As a core algorithm we use this version of fuzzy ants algorithm:

```
1:  randomly distribute ants on matrix area
2:  randomly place objects on area, however mostly one in
    the matrix cell
3:  until finishing conditions are not fulfilled, repeat {
4:        move ant
5:        if ant do not carry any object {
6:              ant can pick any object in the vicinity
7:        }
```

```
8:          if and carry some object {
9:                ant can drop the object in the cell according to
                  the content of the cells in the vicinity
10:         }  }
```

We identify the following input parameters, which are used for modification of behavior of algorithm. These parameters are:

- *size of the matrix* on which are ants moving
- *number of ants* randomly distributed in the matrix
- *number of needles* randomly distributed in the matrix
- *number of needles in the neighborhoods* – used as condition when ant drop the needle
- *type of neighborhoods* or *type of movement* as cross or star
- *number of cycles* of whole algorithm – used as finishing condition
- *number of created piles* in the matrix by ants – used as finishing condition

As a result of this clustering algorithm we get piles of needles of different size at different place in the matrix. We identify piles as cluster and get the information about it.

6 Evaluation

6.1 Ways of evaluation of gained results

For evaluation of gained results we count four measures – precision, recall, F measure and E measure. But because usage of precision and recall separately do not cover success of the system and E measure contains variable parameter, which is not useful to us, we mainly use F measure which presents dependency between precision and recall statistics.

6.2 Reached results

6.2.1 Results for clusteringAnts application

Result of clusteringAnts application output is matrix on which are shown heaps of needles and final position of ants. During the testing of application we discovered that if number of ants is lower, they can also create less heaps but with bigger number of needles. On the other hand, many ants create also many heaps with few needles.

Number of created heaps also depends on the number of needles in neighbor when ants have to drop needle. If number was 1, many small heaps were created. But if we increate this parameter to 2 or 3 needles in neighbor, number of created heaps was smaller but contained more needles.

6.2.2 Results for searchPerson application

Application was tested by using names Anton Balucha, Peter Borga, Miloš Blaško and Pavol Návrat. We evaluate several parameters, which meaning is better to clarify before we provides results:

1. # – order
2. Name and surname – name and surname of person on which we tested application
3. |D| – number of documents which belongs to the person
4. |K| – number of created clusters
5. $|R_i|$ – number of relevant documents about the person for which is created *i* cluster
6. $|I_i|$ – number of identified documents by our application about the person for which is created *i* cluster

7. $|RI_i|$ – number of gained relevant documents, $RI = R \cap I$
8. $Precision_i$ – precision of clustering for *i* cluster and its name
9. $Recall_i$ – recall of clustering for *i* cluster and its name
10. F measure – F statistic counted for cluster
11. Avg. Precision – average precision counted as sum of all precision for the person divided by number of created clusters
12. Avg. Recall – average recall counted as sum of recalls for the person divided by number of created clusters
13. Avg. F measure – average F measure counted as sum of F measures of the person`s all clusters divided by number of created clusters
14. Total Avg. Precision – total average precision counted as sum of average precisions divided by number of names. This value is regarded to be as average precision of whole application.
15. Total Avg. Recall – total average recall counted as sum of all average recalls of all names divided by number of names. This value is regarded to be as average recall of whole application.
16. Total °Avg. F measure – total average F measure counted as sum of all average F measures divided by number of names. This F measure is regarded to be as F measure of whole application.

Results for single names are presented in Table 1 to Table 6. From these results we can calculate total average precision, total average recall and total average F measure:

- total avg. precision is 0.552803 = 55.28 %
- total avg. recall is 0.87125 = 87.125 %
- total avg. F measure is 036004392319 = 60.00 %

Table 1. Number of relevant documents for persons.

#	Name and surname	|D|	|K|	$\|R_i\|$
1	Anton Balucha	8	8	2, 2, 2, 2, 2, 2, 2, 2
2	Peter Borga	10	10	2, 2, 2, 1, 1, 1, 1, 1, 1, 1
3	Miloš Blaško	10	10	8, 8, 1, 8, 1, 8, 8, 8, 8, 8
4	Pavol Návrat	10	10	10, 10, 10, 10, 10, 10, 10, 10, 10, 10

Table 2. Number of identified documents on which should by person according to application.

#	Name and surname	$\|I_i\|$
1	Anton Balucha	5, 7, 8, 6, 6, 5, 6, 5
2	Peter Borga	8, 8, 7, 9, 3, 5, 6, 3, 6, 7
3	Miloš Blaško	10, 9, 8, 8, 10, 5, 10, 10, 8, 8
4	Pavol Návrat	8, 8, 8, 8, 7, 2, 7, 3, 7, 8

Table 3. Number of relevant identified documents.

#	Name and surname	$\|RI_i\|$
1	Anton Balucha	2, 2, 2, 2, 2, 2, 2, 2
2	Peter Borga	1, 2, 2, 1, 1, 1, 1, 1, 1, 1
3	Miloš Blaško	8, 7, 1, 6, 1, 4, 8, 8, 7, 6
4	Pavol Návrat	8, 8, 8, 8, 7, 2, 7, 3, 7, 8

Table 4. Average precision.

#	Name and surname	$\|Precision_i\|$
1	Anton Balucha	0.4, 0.285, 0.25, 0.333, 0.333, 0.4, 0.333, 0.4
2	Peter Borga	0.12, 0.25, 0.28, 0.11, 0.33, 0.2, 0.16, 0.33, 0.16, 0.14
3	Miloš Blaško	0.8, 0.777, 0.125, 0.75, 0.1, 0.8, 0.8, 0.8, 0.875, 0.75
4	Pavol Návrat	1, 1, 1, 1, 1, 1, 1, 1, 1, 1

Table 5. Average recall.

#	Name and surname	$\|Recall_i\|$
1	Anton Balucha	1, 1, 1, 1, 1, 1, 1, 1
2	Peter Borga	0.5, 1, 1, 1, 1, 1, 1, 1, 1, 1
3	Miloš Blaško	1, 0.875, 1, 0.75, 1, 0.5, 1, 1, 0.875, 0.75
4	Pavol Návrat	0.8, 0.8, 0.8, 0.8, 0.7, 0.2, 0.7, 0.3, 0.7, 0.8

Table 6. Average precision, average recall and average F measure.

#	Name and surname	Avg. Precion	Avg. Recall	Avg. F measure
1	Anton Balucha	0.341964	1	0.509647055
2	Peter Borga	0.211468	0.95	0.345932217
3	Miloš Blaško	0.657778	0.875	0.750996883
4	Pavol Návrat	1	0.66	0.795180723

7 Conclusions

On actual work we successfully analysed problem area and possibilities of using different procedures and algorithms. We successfully implemented prototype of web application located on *http://www.tonyb.sk/search.jsp*, which brings results of text processing and clustering. We also implemented second application which simulate clustering of needles be ants. On present work still many work, which needs to be done and needs to be improved. We identify several items which move forward our application and bring better results of searching.

Acknowledgement: This work was partially supported by the Slovak Research and Development Agency under the contract No. APVV-0208-10.

References

[1] Gaceanu, R. D., Pop, H. F.: *An Adaptive Fuzzy Agent Clustering Algorithm for Search Engines*. Department of Computer Science Babes Bolyai University, Department of Computer Science Babes Bolyai University, 2010.

[2] Laclavík, M., Šeleng, M.: *Vyhľadávanie informácií*. Učebné texty k predmetu Vyhľadávanie informácií, 2011. Available at *<http://vi.ikt.ui.sav.sk/@api/deki/files/1024/= vi_text.pdf>* (23.04.2012)

[3] Schockaert, S., Cock, M. D., Cornelis, C., Kerre, E. E.: *Clustering Web Search Results Using Fuzzy Ants*. Department of Applied Mathematics and Computer Science, Ghent University Fuzziness and Uncertainty Modelling Research Unit Krijgslaan 281 (S9), B-9000 Gent, Belgium, 2007.

Facial Expression Recognition for Semantic User Modeling

Máté FEJES*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
matefejes13@gmail.com

Abstract. Facial expressions along with verbal and written communication offer a rich opportunity for expression of ideas. These properties can be utilized in various fields of human computer interaction. This article describes our method proposed to recognize emotional aspects of human face by observing the subject via webcam. Most of similar systems support recognition of some basic emotions such as joy, anger, or surprise. Our goal is to represent the emotional state more sophisticatedly. Our implementation decomposes facial expressions into a set of atomic motions of the face (so-called Action Units), such as raised eyebrows, lip corners pulled or eye blinking. Our method is designed for use in semantic web-based systems. Finally, we proposed a method for user modeling that utilizes emotion recognition as a source of implicit user feedback.

1 Introduction

Human computer interaction covers the methods of information exchange between man and computer. The interaction is typically used to obtain explicit user commands and/or to collect implicit feedback, which is the more problematic of the two. The observations of implicit actions may be ambiguous in that we try to guess what the user is thinking without actually knowing it explicitly. In this paper we explore detecting user's facial expressions/emotions – which ultimately serve as a vehicle for better user modeling – as one of the possible ways to obtain implicit information from the user beyond the scope of the typical human input devices allow.

Basic human emotions and their expressions are innate generic reactions, constituting an implicit way of communication. Implicit signals like tone of voice, gestures and facial expressions are applied in verbal communication and often have non-trivial power of expression, which can confirm, refute or totally alter the meaning of the verbal part of communication. Analogously in the task of information retrieval, users' informational need affects his/hers emotional need, and vice versa [2]. Our project is based on this influence of user's informational and emotional needs, ultimately aiming to enrich the user feedback with them.

In this paper we describe the stages of our research. We propose a method for recognizing facial expressions/emotions of a human subject based on a sequence of images (frames) of the

* Master degree study programme in field: Software Engineering
Supervisor: Dr. Jozef Tvarožek, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

subject's face. In order for the recognition method to provide feature rich input for subsequent machine learning-based method of user modeling, we recognize lower level facial features that can be effectively used to build up the higher level emotions. Most existing recognition systems consider the discrete representation of the six basic emotions: joy, sadness, anger, disgust, surprise and fear [5], while others represent the extracted information in two-dimensional space (positive – negative and active – passive).

Our experiments have shown that the facial expression of these basic emotions can be more complex; consequently we decided to recognize facial features with lower granularity. The output of our method is a set of small atomic movements of the facial muscles – so called Action Units [1] – which are, or are not present in the input image. Due to their physical nature, the complex facial expressions consist of the simple movements. Using this representation we obtain a more accurate description of user's emotional state. Our approach is based on several similar implementations [5, 6] which are realized using Support Vector Machine (SVM) learning.

In the final stage, we propose a user modeling method that uses the emotional states for user/student modeling in a personalized information system, which, in our case, is an online web-based learning environment used by hundreds of users in teaching of programming. Our method determines the relations between the emotion recognition output and user activities within the information system. The ultimate goal is to anticipate user's (student's) immediate action based on previous activities and emotional state.

2 Emotion theory

According to the seminal work by T. A. Wilson [4], there exist various types of people's (or users in case of information retrieval) needs, which affect their behavior. These are:

1. *Physiological need*: basic instinctual needs of people and animals for survival, such as hunger, avoiding danger, sexual desire, and regeneration.
2. *Information need*: desire of an individual or a group to obtain certain information in order to compensate for lack of knowledge.
3. *Emotional need*: desire of an individual to get into a particular emotional state by obtaining necessary information or emotions. Emotional state is a set of emotions that the subject is experiencing at a given time. Emotional need does not have to be limited to positive emotions. There are so called strategic emotions, which may be negative, but motivate people to eliminate failure [2].

The reason for satisfying of human needs by information seeking is the fact that different types of needs are related and caused by each other. Physiological needs are not satisfied by finding the necessary information directly to solve the problem. They lead to a need on a higher level – information or emotional need. Finally, the problem caused by information or emotional need is addressed directly [4]. For example, physiological need like hunger can motivate users to look for restaurants nearby, or quite the opposite, it can lead to frustration, and as a result the user starts to look for entertaining causal content (e.g. funny YouTube videos).

Based on different assumptions, emotional needs are more fundamental than information needs, i.e. a corresponding emotional need belongs to each information need. Obtaining information to solve emotional needs usually cancels out the respective information need. The implication is one-way: the existence of an emotional need does not assume the existence of an information need. Under this assumption the range of emotional needs is wider than the range of information needs [4].

This implies that the behavior of the users of information systems is influenced by emotional aspect to a greater extent. When considering an information need, the relevant emotional need is also present.

3 Facial expression recognition method

In this section we summarize the proposed facial expression recognition method. We describe the dataset, preprocessing, the different approaches we explored, and the evaluation of the method.

Training dataset – a representative set of images for the purpose of training is required. The requirements on the training data are: cataloged by Action Units, a large number of subjects, men and women preferably in equal proportions, good quality, and realistic pictures. Due to these requirements, we obtained the *Bosphorus* database [3] that consists of 4666 images. The use of database was negotiated with its authors; we did not find a suitable larger dataset.

We preprocessed the dataset with *Luxand Face SDK* library. Facial features are attributes of the face that are characteristic to facial expressions. These properties are defined by simple movements of different parts of face (e.g. eyebrows, mouth). The library can recognize a human face in an image and determines the positions of 66 different facial points that constitute the basic structure of the face. The library's recognition performance is fast and accurate.

3.1 Classifiers

As we discussed above, different facial expressions are recognized by capturing of certain typical muscle movements. Furthermore, the positions of facial points can be used to describe the various movements (see Figure 1). To capture the movements (change in the position of points), we have to devise a suitable representation for each state. States can be represented by the relative positions of the facial points. The representation is effectively a vector of real numbers, in which each value (dimension) can be expressed either as distance or as height difference of two points or as angle of a triplet of points. Such a vector is also called a *feature vector*.

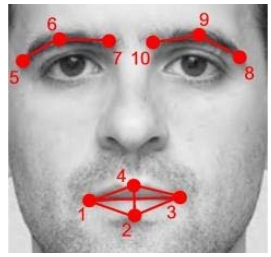


Figure 1. Sample classifier for smile facial expression.

Figure 1 shows a simplified classifier example. The example is designed for monitoring smile. To recognize this movement we should take into account the distance of points 1 and 2, the height difference of points 1, 2 and 3, 2 and also the angle enclosed by the triplet of points 1, 2 and 3. In this case, the vector that describes this situation will be as follows:

$$v = \big(d(1,2), \quad \Delta h(1,2), \quad \Delta h(3,2), \quad ang(1,2,3)\big)$$

where $d(x,y)$ is the distance of points x and y, $\Delta h(x,y)$ is the height difference (difference of y coordinates) and $ang(x,y,z)$ is the angle enclosed by triplet of input points (the second input point is the vertex). Distances are normalized – horizontal distances are divided by the distance of eyes, vertical distances by the height difference of eyes and nose tip.

3.2 Process outline

Based on the observations above, we designed and implemented a method for recognizing emotions. This method consists of two separate phase – training and recognition (testing). In training, we process the training subset (60 % of all images) cataloged by different classes. For each image we extract facial features and create a vector (or more vectors) by which the emotional

state is characterized. Machine learning is performed by using SVM. The output of the training is the SVM model. In recognition (testing) phase, the input images are processed without the any information about the class (or classes) they belong to. Again, we start with extracting of facial features and then we create a recognition vector for each image. Finally, the recognition vector is compared to the training vectors (produced in the training phase), which are available in the SVM model. SMV provides output in the form of percentages for each class.

3.3 Proposed approaches

In order to achieve better recognition accuracy we have done more experiments to explore different approaches. The fundamental difference between the approaches is mostly related to the number and granularity of classes and also to classifiers, i.e. vectors that represent the various states. We discuss some of the more interesting experiments and results:

Approach 1 – Large granularity and one generic classifier: In this experiment, we chose a large granularity of the classes (relatively small number). All examined points (facial features) were relevant to each class. We represented each state (still image) by one vector that was the same for each class. Samples were divided into 6 classes. These classes correspond to six basic emotions by discrete view of Paul Ekman [1]. We used a generic classifier, i.e. representation of the state by one common vector for each class. In this case, the vector was formed by the distances between each pair of points, or lengths of edges in the complete graph, which has 66 vertices (2145 values).

Approach 2 – Large granularity and one specific classifier: This experiment was similar to the previous in term of classes and the number of vectors. We used the distribution for 6 emotions and state representation by one vector for each class, but this time we have defined the vector values manually. Again, we used a common classifier, i.e. one vector for each image and each class. The values of the vector were chosen specifically. Each value was focused on a particular part of the face, or a specific movement, we want to capture.

Approach 3 – Small granularity and more specific classifiers: We have come to believe that the facial expressions can be much more complex, so they should not be represented by one general object. We proposed new classes of smaller granularity, each of them focusing on a specific part of the face. We split the face and its movements into the basic units called Action Units [1]. We selected 8 units that are most common and recognizable from ordinary photos: inner brow raiser (AU 1), outer brow raiser (AU 2), brow lowerer (AU 4), lip corner puller (AU 12), dimpler (AU 14), chin raiser (AU 17), lip stretcher (AU 20), lip tightener (AU 23) [1]. Each class relates to a different specific part of the face, so for each of them we have designed a classifier (feature vector), which describes the given part. Each image is represented by 8 vectors, each of them expressing the probability of the presence of the particular Action Unit within the image.

3.4 Evaluation and discussion

Evaluation of all approaches was performed by an identical method. We have trained the selected classifier for a 60% subset of all images. A 20% subset was used as the validation set, i.e. in these images we were trying to improve the recognition (testing) accuracy with different parameters of SVM training. We have applied the test method over the remaining 20 % of the images (test set). The results were grouped into two values. Average True Positive (AvgTP) is the average of results (similarity percentage) of comparing the images to a class, which they really belong to. Average False Positive (AvgFP) is the average of results of comparing the images to a class, which they does not belong to. In our case the tests were performed over sets of either 100 % positive or 100 % negative samples, consequently, true/false negative values (AvgTN, AvgFN) are always complementary to the positive (TP + FN = FP + TN = 100 %).

We set a goal to get the AvgTP value over 50 %, considering that the recognition is typically performed on a stream of images. The 50% accuracy provides a reasonably accurate recognition rate. Table 1 contains a summary of these experiments.

Table 1. Recognition results for the proposed approaches.

	AvgTP	AvgFP	AvgTN	AvgFN	σ TP	σ FP
Approach 1	35,2 %	9,7 %	90.3 %	64.8 %	14,5 %	8,9%
Approach 2	37,7 %	13,4 %	86.6 %	62.3 %	11,2 %	8,2%
Approach 3	68,5 %	36,2 %	63.8 %	31.5 %	11,1 %	12,3%

The first approach has failed to meet the 50% accuracy threshold. Regardless of the size of result value, the difference between them plays an important role. Although the difference between the average accuracies is sufficiently large, the TP and FP values were often very similar due to the standard deviation, so we considered this attempt a failure.

The second approach brought a little improvement compared to the first. Standard deviation of both values was less than before. Regarding to our aim and the difference between the values of this approach, we again declared this approach a failure.

The final third proposed approach of training and recognition brought some visible improvements. AvgTP value exceeds the specified minimum 50 %, the difference of values AvgTP and AvgFP in this case is the largest. Standard deviation is around previous levels, there has not been a major change.

The complete graph of distances of face points in the approach obviously did feed too many unimportant inputs into the classifier rendering it ineffective. Manual classes according to the face structure brought in improvements, mostly due to “more contrast” in fewer data points that were processed by the classifier. Finally, selecting the most differentiating facial features did generate best results in the third approach.

4 User modeling based on facial expression recognition

In the second stage of our research, we aim to employ the emotional recognition for user/student modeling in a personalized information system. We assume a typical system, in which user performs various activities such as information retrieval, studying, problem solving, or even playing games. We further assume that the system supports logging of these activities and contains "classical" user model. In such a system the information needs are determined by monitoring of user actions. Provided the assumptions about information and emotional need (see Section 2) are valid, we are able to estimate the need for information from the emotional state.

In order to do so, we target an online web-based learning system *Peoplia*. We have gathered activity logs on several hundreds of users/students working within the system. The logs are currently analyzed for user action sequences. Furthermore, we have extended the system with a webcam component that captures user/student face with the rate of 8 to 10 frames per second. As the next step we plan on to carry out an experiment in which we would monitor the information (based on the user actions) and emotional needs (based on the webcam imagery) of users during the session simultaneously. Then, we will enrich the discussed user model with the obtained emotional states. In the extended user model each user activity has a corresponding emotional value. This would be followed by comparing corresponding groups of activities and emotions. The aim of the experiment is to find certain patterns of behavior and psychical reactions to them. We aim to estimate the user’s immediate next action using his/her previous actions and the current emotional state.

Figure 2 demonstrates the approach. The captured frames provide us the AU levels at any given time (perceptual values of AU1, AU2 and AU3).We will aggregate the collected data according to a pre-specified window length. Following this, we assign two values to each interval: a characteristic activity from the logs (e.g. Activity A) and a vector containing the average levels of Action Units within the interval (e.g. x_1, y_1, z_1). Finally, we use machine learning to train the system. We consider classes by the list of activities and classifiers by the vectors discussed above.

The outcome of this training is going to be a model that for an emotional state assigns a possible immediate user action/activity. In other words, we observe users via a webcam and using this method we estimate theirs immediate next steps in real time.

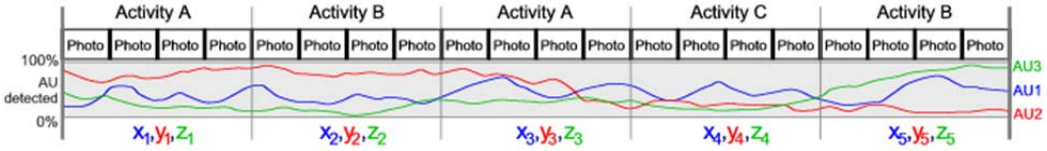


Figure 2. Comparison of user activities to emotional state.

5 Conclusion

In this paper we proposed and evaluated various approaches to emotional state recognition based on facial expressions. We tried to achieve better accuracy, so we did more experiments with different approaches. The best performance was 68.5 %, which we consider to be sufficiently accurate for recognition from a stream of images. Do note that our proposed method uses only general calibration, that is, it is trained on a dataset of multiple different subjects. Training the classifiers for each individual user separately is a different class of methods, and would bring even higher accuracy in the expense of more obtrusive user experience; we did not explore this approach tough. In the second stage of our research, we propose a user modeling method that uses the recognized emotional states for modeling user/student in a personalized information system.

Acknowledgement: This work was partially supported by the Slovak Research and Development Agency under the contract No. APVV-0233-10.

References

[1] Ekman, P., Matsumoto, D. R., Friesen, W.V.: *What the Face Reveals: Basic and Applied Studies of Spontaneous Expression Using the Facial Action Coding System (FACS)*. New York. Oxford University Press. (1997).

[2] Moshfeghi, Y.: *Role of emotion in information retrieval*. PhD thesis. University of Glasgow, (2012).

[3] Savran, A. et al.: Bosphorus Database for 3D Face Analysis. *The First COST 2101 Workshop on Biometrics*. Denmark. Roskilde University, (2008).

[4] Wilson, T. A.: On user studies and information needs. *Journal of Documentation, 30(1)*. (1993), pp. 3–15.

[5] Kotsia, I., Pitas, I. Real time facial expression recognition from image sequences using support vector machines. *IEEE International Conference on In ICIP 2005*. 2005, vol. 2, (2005), pp. 966–969.

[6] Shan, C., Gong, S., McOwan, P. W. Facial expression recognition based on Local Binary Patterns: A comprehensive study. *Pattern Recognition and Image Analysis*. (2009), vol. 17, pp. 592–598.

What Makes the Best Computer Game? How Game Features Affect Game Play

Peter KRÁTKY*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
kratky.peto@gmail.com

Abstract. Personality has an impact on user's behaviour in information systems, and adaptive systems that model these features can provide better user experience. Adding fun to the mix is making games even more intriguing objects of study. In our project we explore user profiling options of computer games by examining how game features and personality traits affect user (player) engagement and game play. We designed a modular casual browser game which enables us to exhaustively study effects of different game mechanics on game play. The game tracks both low-level user interface interactions and high-level game actions. We conducted an extensive study in which we collected data on several hundred game sessions. In this paper we describe our approach and present preliminary results.

Amended version published in Information Sciences and Technologies Bulletin of the ACM Slovakia, Vol. 5, No. 2 (2013), pp. 36-38.

* Master degree study programme in field: Software Engineering
Supervisor: Dr. Jozef Tvarožek, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

Personalized Recommendation of Learning Resources

Jozef LAČNÝ*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
lacny08@student.fiit.stuba.sk

Abstract. Nowadays large amount of information is offered to the user via various information systems and e-shops. Therefore selection of useful information is very important for the user. In this work we propose a method for recommendation of learning resources for groups in the Web. We based our method on existing research done in educational system ALEF at the Slovak University of Technology in Bratislava. We extend it by using the users' learning styles to enhance the suitability of recommended resources by adapting the group creation process and recommendation itself to the users' preferences based on their knowledge and learning style.

1 Introduction

Personalized recommendation plays important role in wide variety of fields nowadays. Its main purpose is to deliver the most relevant content to each user in specific scenario. The library users would like to get recommended books according to their taste, the researchers would like to get papers in their field of study and the shoppers like to be offered goods to buy according their actual shopping purpose. It has been done a lot of research in this field including various recommendation techniques – content based recommendation, collaborative recommendation and many others approaches combining these two or inventing other new methods [1].

An interesting field for personalized recommendation arises in the field of education, where it is important to recommend study materials to achieve better study results and enhance whole learning process. This domain is very specific mainly regarding the process of choosing relevant study materials for recommendation, because the main focus here is not to fulfil individual satisfaction of the user but to help him to learn more effectively and achieve better results in shorter period of time. To take it even further, it is very challenging to recommend for groups in collaborative learning. Collaborative learning helps the students to better understand of the subject of study by letting them to interact and share their thoughts [2]. In collaborative learning it is easier to cover bigger areas of study and advance faster as in individual learning, because diversity of individuals' knowledge in the group makes it necessary to discuss various matters and therefore enhance the whole group's understanding [3].

* Master degree study programme in field: Software Engineering
Supervisor: Michal Kompan, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 37–42.

There is wide diversity in users' learning styles and a lot of research has been done to adjust the learning process to users' learning styles [4]. In this paper we propose a novel method for personalized recommendation of learning resources for groups incorporating users' individual learning styles. To choose the right resources to recommend we utilize the users' actual knowledge, knowledge of prerequisites and his learning styles which together create main criterion to find and recommend appropriate learning resources.

2 Related work

There are several approaches to personalized recommendation in adaptive educational systems. The research is mainly focused on the identification of student's preferences for purpose of recommendation appropriate learning resources. Various recommendation methods are used – there are content based approaches [5], collaborative methods [6], and their combinations or other hybrid systems [7].

Solution based on learners' role-based multi-dimensional collaborative recommendation [6]considers students' activity as a sequence of actions that user makes while interacting with the system. It divides the students into two groups (roles) using Markov chain – beginners and advanced learners. These two roles together with explicit learning object rating are the basis for recommendation, while weight of rating of advanced learners is higher.

System AHA! (Adaptive Hypermedia Architecture) [8] provides adaptive content and adaptive navigation to students in e-learning environment. It supports adaptive content and adaptive navigation. It uses layered user model that stores information about user knowledge and his interaction and supports knowledge spreading to related concepts. User model is then refreshed while interacting with the system and further used to adapt content and navigation regarding to defined rules.

ALEF [5] is an adaptive educational system developed and used at the Slovak University of Technology in Bratislava. Its domain model consists of learning materials and their metadata which are connected to each other. ALEF provides three kinds of learning objects: questions, explanations and exercises. In [9] the authors proposed an extension of this system with personalized recommendation of learning objects for single user considering limited time of learning. Users' target knowledge of particular subject is set before the learning and learning objects are recommended in purpose to help the student to learn a defined set of concepts in a given time to a given level.

The method assumes that it is better to learn more concepts partially than just few concepts poorly (in case we have limited time to learn). When evaluation the objects suitable for recommendation it takes into account thematic suitability of object, difficulty suitability and object repetition suitability.

All mentioned solutions use some kind of personalisation in the process of learning but none of them considers in their recommendation strategy users' cognitive styles that can be a valuable asset in personalizing the learning process. ALEF is designed to be easily extensible by various modules and therefore we have chosen it to implement and test our method.

3 Recommendation of learning resources

In order to recommend relevant learning resources to students in adaptive educational system there are several matters to consider. First to determine student's preferences we need to create a credible user-model, next we evaluate available learning resources with taking into account the user-model and recommend them. In recommendation for groups there is also important step of division of the users into groups and building the group recommendation.

3.1 User model

The starting point for recommendation in any form is well formed user-model. In this case the user-model gathers information about the student's knowledge, interaction with educational system and his explicit feedback. Further recommendation is based on these characteristics. User-model for modelling user knowledge in our method is based on computer-adaptive testing (CAT) [10] which has been extended in [11] with certainty factor to store the certainty that user gains some knowledge.

We have extended this model with student's learning styles developed by Silverman and Fedler [4], that describe cognitive style in four dimensions: perception (sensing / intuitive), input (visual / verbal), processing (active / reflective) and understanding (sequential / global). To get the students' learning styles we incorporated an adaptive hierarchical questionnaire [12] which introduces new approach to predict students' learning styles by reducing the number of questions of original questionnaire presented by Index of Learning Styles [4] (from 11 to 4-6 questions per dimension). The student's learning style is then defined by a vector (see equation 1) of four values corresponding the four dimensions of learning styles in ranges from $\langle 0,1 \rangle$.

$$user\_learning\_style = \{per\_dim, inp\_dim, proc\_dim, und\_dim\} \tag{1}$$

3.2 Evaluation of learning objects

Purpose of evaluation of learning objects is to pick an object (exercise, question or learning text) that is most suitable to achieve the goal of learning – cover and understand the studied subject in a sufficient level. We derived our method from [9] that utilizes three criterions when evaluating learning objects as follows:

1. Suitability of concepts (includes prerequisites fulfilment and student's knowledge evaluation).
2. Suitability of object difficulty (based on CAT [10]).
3. Repetition of recommended objects.

Each of mentioned criterions is represented as value in range $\langle 0,1 \rangle$ and the purpose is to find an object, that satisfies them best. There are two important kinds of relations between learning objects to be considered when performing recommendation: 1. relations between concepts including generalization (its weight is always 1), prerequisite relation and concept connection and 2. relations between objects and concepts, where each learning object is defined as vector of weights indicating its relation to particular concept.

3.3 Preference prioritization

Our method extends previously mentioned learning object evaluation approach with the learning styles' influence in the calculations. In [4] the authors described students preferences prioritization in the learning process regarding their learning styles. We consider this when evaluating learning objects in following manner:

1. Concept difficulty tolerance: this criterion is considered when evaluating concepts' recommendation suitability. We adjust parameter K in equation 2 regarding user's learning style dimension of understanding, where we decrease the concept's suitability for sequential learners.

$$C = \frac{1}{1+10^{9+(knowledge\_level-1)}} * K \tag{2}$$

2. Prerequisites fulfillment: in this calculation we adjust the parameter L in equation 3 regarding the users learning styles in perception dimension. It modifies the final value of the prerequisites fulfillment criterion regarding user's learning style. We decrease the concept suitability for sensing learners and increase it for the intuitive learners.

$$P = min(1, knowledge\_level - relation\_weight + 1) * L \quad (3)$$

3. Object difficulty tolerance: in this criterion we adjust the parameter C in equation 4 regarding the user's learning style dimensions of understanding and processing. By widening or reducing of the Gaussian curve we can simply adapt the object's suitability regarding user's learning style. The sequential learners have slightly higher and active learners have higher tolerance to object difficulty so we adjust the parameter C to fit the user's preferences.

$$O = e^{-\frac{(object\_difficulty - knowledge\_level)^2}{2C^2}} \quad (4)$$

4. Preference of specific objects: we adjust the relevance of recommended objects regarding user's learning style, e.g. if it prioritizes exercises before questions we change the priority (and therefore order) of recommended objects. The visual learners prefer learning objects containing symbolic explanations in form of graphs and pictures, on the other side verbal learners prefer written explanations.

The real values of the constants used in the learning object evaluation process will be determined from the experiments.

3.4 Group recommendation

We create temporary groups from students actually logged in the educational system and actively interacting with it. The recommended number of students in a learning group is 4-7 students [4], to our purposes we use groups of 4 or 5 students. Table 1 illustrates the group distribution process.

Table 1. Distribution of students in groups.

Number of users	Distribution
4	1x4
9	1x4, 1x5
10	2x5
12	3x4

We have chosen this limitation, because assumption that the more students in group, the bigger diversity of their learning styles. To divide online users into groups we use clustering algorithm k-means, to create groups containing students with similar learning styles.

As we have picked some learning objects from individual users in previous step (in order set by suitability, difficulty, knowledge and learning styles) next we need to merge these recommendations for recommendation to groups. To aggregate recommendations in a group, when the recommended objects have low variance (high level of consensus) it is suitable to use the least distance heuristics to set priority of a learning object. On the other hand, when the recommendations have high variance it is more suitable to use average value heuristic [13]. To handle with both mentioned cases we will use hybrid approach which will use both heuristics regarding the value of standard deviation (equation 5).

$$recommendation = \begin{cases} least_{distance}, & std_{dev} < B \\ average_{value}, & std_{dev} \geq B \end{cases} \quad (5)$$

The value of parameter B will be set experimentally.

4 Evaluation

We have implemented our method using framework Ruby on Rails as a widget of existing educational system ALEF [5].

We plan to test the method live on real users during picked courses of semester. We intend to verify two aspects of our solution:

1. recommendation,
2. learning.

Recommendation will be verified in two steps:

1. a priori: using implicit users' feedback (whether the users really follow the recommendations),
2. a posteriori: using explicit users' feedback (evaluation of the recommendation by users).

When verifying learning we will divide students into three groups: users without any recommendation, users with individual recommendation and users with group recommendation. All groups will use the educational system during some time with or without recommendation. We will evaluate the results of experiment using quantitative measures:

1. pre- and post- tests for all groups to determine change of knowledge,
2. automatic evaluation of user-knowledge changes in groups.

We will evaluate learning with qualitative measures too using short questionnaire in which users will express the asset of group recommendation in learning process.

As a part of evaluation of learning process we have proposed the questionnaire to a selected group of students to determine that there really is variety of learning styles in the future testing group. Results of this experiment show wide variety of learning styles in the testing group and furthermore we were able to easily divide them into studying groups with similar learning styles. First experiments with single user recommendation show promising results for students that already have some system interaction history, but poor results when cold-started. Therefore we will consider postponing the recommendation process after the student interacts with the system for a while.

5 Conclusions

Collaborative learning in general can be defined as any kind of group learning in which there are some meaningful learning interactions between learners [14]. Therefore it is very important to support communication within the students' group. The communication is even more important when dealing with collaborative learning in online environment. In our method we offer an option of online communication by providing simple chat widget to enhance the learning process.

The result of our work is a method for group recommendation in educational systems utilizing users' learning style in the process of group creation and recommendation itself. We use layered user-model reflecting his knowledge and enhance it with his learning styles. The learning styles are then used to prioritize student's preferences in process of calculation of learning resources' suitability for recommendation. In the end we use hybrid approach to aggregate single students' recommendations for group recommendation. Because communication in collaborative learning is very important we take this matter into account and help to allow the students communicate within their group online.

There are many possibilities for future research in this field, especially in exploring other properties of learning resources to be affected by students' learning styles. Other possible extension of our solution is to enhance the group-creation process with taking into account not only students' learning styles but their knowledge as well (e.g. to build groups with tutors).

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/0675/11.

References

[1] Boratto, L., Carta, S.:State-of-the-Art in Group Recommendation and New Approaches for Automatic Identification of Groups, *Information Retrieval and Mining in Distributed Environments*, pp. 1–20, (2011).

[2] Stahl, G., Koschmann, T., Suthers, D.:Computer-supported collaborative learning: An historical perspective, *Cambridge handbook of the learning sciences*, pp. 409–426, (2006).

[3] Yonghui, C. A. D.:Study of Antecedent and Consequences of Work Group Learning, *Science And Technology*, pp. 168–171, (1997).

[4] Silverman, L. K., Felder, R. M.:Learning and teaching styles in engineering education, *Eng. Educ*, vol. 78, pp. 674–681, (1988).

[5] Šimko, M., Barla, M., Bieliková, M.:ALEF: A Framework for Adaptive Web-Based Learning 2.0, *Advances in Information and Communication Technology*, vol. 324, pp. 367–378, (2010).

[6] Wan, X., Ninomiya, T., Okamoto, T.:A Learner's Role-based Multi Dimensional Collaborative Recommendation (LRMDCR) for Group Learning Support, *Conference on Neural Networks*, pp. 3912–3917, (2008).

[7] Zakrzewska, D.:Building Group Recommendations in E-Learning, *Proceedings of the 4th KES international conference on Agent and multi-agent systems: technologies and applications*, pp. 391–400, (2010).

[8] De Bra, P., Aerts, A., Berden, B., Lange, B. De: AHA! The adaptive hypermedia architecture, In *Hypertext and hypermedia*, (2003), vol. 4, pp. 81–84.

[9] Michlík, P., Bieliková, M.:Exercises recommending for limited time learning, *Procedia Computer Science*, no. 2, pp. 2821–2828, (Jan. 2010).

[10] Linacre, J.:Computer-adaptive testing: A methodology whose time has come, *MESA Memorandum No. 9*, no. 69, (2000).

[11] Unčík, M.:Visualization of User Model in Educational Domain, *Proceedings of Informatics and Information Technologies IIT.SRC 2012 Student Research Conference*, (2012).

[12] Ortigosa, A., Paredes, P., Rodriguez, P.:An adaptive hierarchical questionnaire based on the Index of Learning Styles, *Authoring of adaptive and adptable hypermedia*, (2008).

[13] Ninaus, G.:Using group recommendation heuristics for the prioritization of requirements, *Proceedings of the sixth ACM conference on Recommender systems – RecSys '12*, p. 329, (2012).

[14] Laister, J., Kober, S.:Social Aspects of Collaborative Learning in Virtual Learning Environments, *Proceedings of the Networked Learning Conference Sheffield*, (2002).

Discovering and Predicting Human Behaviour Patterns

Štefan MITRÍK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
stevo.mit@gmail.com

Abstract. Large spatio-temporal datasets with trajectories representing human movements can be used to mine behaviour patterns. We show how to transform real-world raw GPS trajectories into the meaningful semantic patterns using combination of the well-known mining and clustering algorithms. We introduce enhancements to the current methods and discusspros and cons of several semantics sources. We present a method that predicts human behaviour in the near future according to the previous behaviour. In addition to that we introduce time degradation that assures that recent changes of the human behaviour are appropriately reflected into the behaviour patterns. We evaluate these enhancements on the real-word data and discuss their benefits.

1 Introduction

The fast development of advanced mobile technologies opens up new possibilities for analysis of humans' behaviour. Location-acquisition technologies such as GPS in combination with intelligent mobile applications allow us to collect huge spatio-temporal datasets of human mobility. These datasets contain trajectories that are performed by individuals during the day. Each trajectory is determined by sequence of visited geographical points and corresponding timestamps.

These datasets give us the opportunity to discover movement behavior and form users' behavior patterns. Each pattern consists of visited locations and routes among them. It also contains time and distance annotations that describe users' behavior in the more detailed manner. A pattern location is enriched by additional semantics information. This transforms geographical points determined by latitude and longitude into the more meaningful places with information about the place semantics such as *University* or *Restaurant*.

The behavior patterns are naturally being performed in repetitive manner. We utilize this to predict the actions of the users in the future. The ability to predict users' actions is crucial in fields such as physical activity recommendation, where we need to recommend the activities in advance so the users can adjust their schedules and plans.

* Master degree study programme in field: Software Engineering
Supervisor: Professor Mária Bieliková, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 43–48.

Even though people naturally repeat similar behavior patterns over and over again, the humans' behavior changes over the time. It can be caused by different year season or change of the timetable at the university. Prediction of the patterns should take this into the account.

Another important thing is effectiveness of the used algorithms. Despite the fact that the performance of the mobile devices grows at the very fast pace there still are some limitations that needs to be considered. One needs to be very careful about the memory consumption because of the platform specific limits of the memory usage.

2 Related work

Several research groups address the problem of behavior pattern discovery and analysis. There is a system that logs GPS trajectories throughout the whole day and discovers important locations that the user visits [1]. With important locations, current position of the user and her previous actions authors of this work designed a method that tries to predict user's future movements. The prediction method is based on Markov Models. They experimented with different orders of the Markov Models, however because of the relatively small size of the dataset (data collected over period of 4 months) they chosen second order Markov Models. The prediction method works on the fly, so the prediction of a next location is performed when user actually is in certain location. While this may be appropriate in some cases, there are cases where prediction is needed in advance. The authors identified a problem with change of the user behavior such as end of the semester. These changes are reflected into the model very slowly. They suggest to use different weights to evaluate behavior patterns and thus favor more recent activities.

Another work deals with tracking of user activity with user's smart phone [6]. Tracking systems need to be as much energy efficient as possible because the sensors that are being used to monitor human activity consume huge amount of the phone battery. They try to solve this problem by constructing user's behavior pattern and predicting future movements. Thanks to it, they know when to turn on and off tracking sensors. The prediction method is similarly based on the first and second order Markov Models and predicators based on *LZalgorithms*.

Even though systems for human movements prediction already exist, they don't deal with prediction in advance that can be very important in some situations. Change of users' behavior is another thing that needs to be considered as human behavior naturally changes in time.

3 Method for discovering and predicting patterns

We propose a method for transform raw GPS coordinates to the meaningful semantics patterns. The process of discovering patterns consists of several steps. We firstly discover important locations that are frequently visited by the user and acquire their semantics. After that we construct the behavioral patterns and annotate them with time and distance info.

With semantics patterns and knowledge about user's previous actions, we are able to predict the pattern that will occur in the future. We consider in our method time degradation that affects both creation and prediction of the patterns so our method reflects changes of human behavior.

3.1 Cluster mining and acquiring semantics

Each trajectory consists of time-stamped sequence of geo-coordinates. The first and the last geo-coordinate of the trajectory represent initial and terminal location of the trajectory because users usually acquire/loose GPS signal when they enter/leave the buildings. A *location* basically represents a place where user has spent some time between her moves.

People usually visit the set of the same locations repeatedly. The process of mining clusters identifies frequently visited geo-locations. We take advantage of the modification of ESDC: efficient density-based subspace clustering [2] that divides the space using density conserving grid

and build hyper-cubes. Each hypercube encloses potential clusters so repeated expensive database scans are avoided. ESDC assures completeness so none cluster is pruned.

All clusters are represented by latitude, longitude and number of visits. However the semantics information is needed (we mean place category by semantics information). We experimented with three different semantics sources. The summarization of their characteristics is shown in Table 1. Note that the number of locations and sufficient semantics information was mainly analyzed in the geographical area of the Slovakia but roughly represents the current state in the western countries.

Table 1. Characteristics of the semantics sources.

	Google Places	Open Street Maps	Foursquare
Offline access		*	
Simple to use API	*		*
Sufficient number of locations	*		*
Sufficient semantics information		*	*

Despite our best effort to automate the process of semantics acquisition there still are some situations where manual corrections are needed. This is especially true in densely build up areas, with many locations standing next to each other. Because of the GPS imperfections we are not able to distinguish locations that are close to each other such as shops or cafes in the malls.

Straightforward solution to this problem is to ask the user about the cluster semantics. However the process needs to be made as easy as possible because users naturally don't like to manually enter information. In our implementation we show the location on the map and ask the user to choose appropriate location category. We order the categories according to the data from Foursquare so the appropriate category is usually among the top of the choices list.

3.2 Construction of the patterns

With clusters accompanied by semantics information and user's trajectories, we are able to construct behavior patterns. A *pattern* represents the sequence of visited locations and transitions among them performed during one day. An example of such pattern:

$$Home_{7:30} \xrightarrow{300/1800m} University_{12:20}^{8:15} \xrightarrow{420/420m} Library_{18:45}^{14:15} \xrightarrow{300/2350m} Home^{19:20} \quad (1)$$

The upper right time annotation represents the time when user entered into the location and similarly the lower right time annotation represents the time when user left the location. The distances above the arrows represent the distances user walked and total distances user passed between the two locations (including distance passed in car or bus).

When searching for frequent patterns we need to take *subpatterns* into consideration. *Subpattern* is a pattern that consists of a subset of *superpattern's* locations that are visited in same order. We show example of the pattern (2) and two *subpatterns* (3,4).

$$A \rightarrow B \rightarrow C \rightarrow D \rightarrow A \quad (2)$$

$$A \rightarrow C \rightarrow A \quad (3)$$

$$B \rightarrow C \rightarrow D \rightarrow A \quad (4)$$

Pattern mining is computationally expensive problem as there is huge number of possible *subpatterns*. We employ PrefixSpan [7] *(Prefix-projected Sequential Pattern Mining)* algorithm that is not only effective but also assures completeness. Its main idea is to examine only prefix subsequences and project only their corresponding postfix subsequences into the projected

database. Thanks to this, we examine only potentially frequent patterns. A pattern is considered to be frequent, when it occurs more than *min_support* times in the given input sequences.

In addition to that we introduce another parameter called *max_jump* that specifies how much can *subpatern* differ from *superpattern*. This can be used to filter out unrealistic subsequences that consist only of few frequent locations. We annotate the patterns not only with time annotations [4, 5], but also with distance annotations that represent distances user passed between the locations.

3.3 Pattern prediction

Prediction of the user behavior can be very useful in some fields such as recommending systems. One may adjust recommendations so they do not collide with user program or plans with knowledge about future user actions. We need to stress out that we deal with predictions so there is no guarantee that the user will actually behave in predicted way.

Our prediction method is based on the fact that most people do repeat similar behavior with respect to the physical activity and transfers among the geo-locations. The prediction method estimates user's future actions according to the actions she made in the past. For example if someone visited the same set of locations in the same order during the last two Wednesdays we may expect that she will behave similarly also during the next Wednesday. We believe that people performs similar actions on the *day of the week* basis so when we try to predict user actions for Wednesday we analyze data from passed Wednesdays.

The prediction method basically evaluates all the past patterns and the pattern with the best evaluation is chosen as a prediction. Please note that the method evaluates not only all the patterns but also all of their *subpatterns* so no frequent possible combination is lost. We consider two things during the patterns evaluation:

- Probability
- Coverage

Probability represents the certainty with which will pattern occur in the future and coverage represents the ratio between the length of the examined pattern and average pattern length. The final evaluation is then calculated as a multiplication of probability and coverage. We may better explain our method on the short example. Lets say we have two patterns (5,6) that were performed during the same day of the week.

$$A \rightarrow B \rightarrow C \rightarrow D \rightarrow A \tag{5}$$

$$A \rightarrow E \rightarrow C \rightarrow D \rightarrow F \rightarrow A \tag{6}$$

When evaluating all patterns and *subpatterns* there will be pattern *(A, A)* that has the value of probability set to 1.0 because it occurs in all patterns. However the coverage value is only 2/5.5 = 0.36 and thus the final evaluation for this pattern is calculated as 1 * 0.36 = 0.36. On the other hand there is a pattern *(A,C,D,A)* that has also probability set to 1.0 and the coverage is 4/5.5 = 0.73. This pattern has highest evaluation and thus is chosen as a prediction for the user behaviour.

3.4 Time degradation

The human behavior changes in time. It can be caused by different year season for example some people may walk more during the summer than winter or many other factors such as holidays or changes of the schedule. In our method we favor more recent behavior of the user. However, we need to be careful about the behavior anomalies and special situations such as sickness or one day holidays that don't have repetitive character. We use weighted average with coefficient *time_deg* that affects how much does the pattern influence prediction and also time and distance annotations.

Assuming we have N previously measured items $z_1, \ldots, z_N$ where z_1 denotes the oldest and z_N the newest one. We calculate the prediction for item z_{N+1} with time degradation *(td)* as follows:

$$z_{N+1} = \frac{1*z_1+td*z_2+td^2*z_3+\cdots+td^{N-1}*z_N}{1+td+td^2+\cdots+td^{N-1}} = \frac{\sum_{i=1}^{N} td^{i-1}*z_i}{\sum_{i=1}^{N} td^{i-1}} = \frac{\sum_{i=1}^{N} td^{i-1}*z_i}{\frac{1-td^N}{1-td}} \quad (7)$$

Imagine a situation where a user moved between the locations AB every Tuesday during the last three weeks with following distances: 100 m, 200 m and 300 m. The prediction would be 200 m (average) for next week without time degradation. However, with time degradation we favor the more recent transitions over the older ones. With *time_deg = 2* the prediction for next week would be 242,8 m *(100*1 + 200*2 + 300*2^2)/(1 + 2 + 2^2)*.

The bigger the *time_deg* coefficient is the more weight is put on recent behavior. Selection of *time_deg* parameter basically is a process of searching for balance between the anomalies filtering and discovering real behavior changes.

4 Evaluation

We evaluated our methods with the combination of real-word GPS trajectories and data that were generated with respect to the real-world data. The introduced methods are programmed in Java language and were tested on various Android devices. We created a module that is integrated into the Android fitness app called Fitly *(formerly known as Move2Play)* [3] that is source of our real world data.

Specifically we analyzed data from 3 users that were collected over the period of at least 6 months. The process of pattern creation was evaluated manually by comparing the discovered patterns and trajectories on the map. We found out that the quality of discovered patterns highly relies on the input data.

The most innovative part of our work is pattern prediction and time degradation. While prediction of repetitive patterns is pretty straightforward, the challenge lies in filtering anomalies and unexpected situations. Based on real world data, we have identified three different situations that need to be considered:

1. One day anomalies *(i.e.: sickness, traveling, national holiday)*
2. Longer anomalies *(i.e.: Christmas time, summer holiday)*
3. Dramatic change of behavior *(i.e.: end of semester, end of season work, move house)*

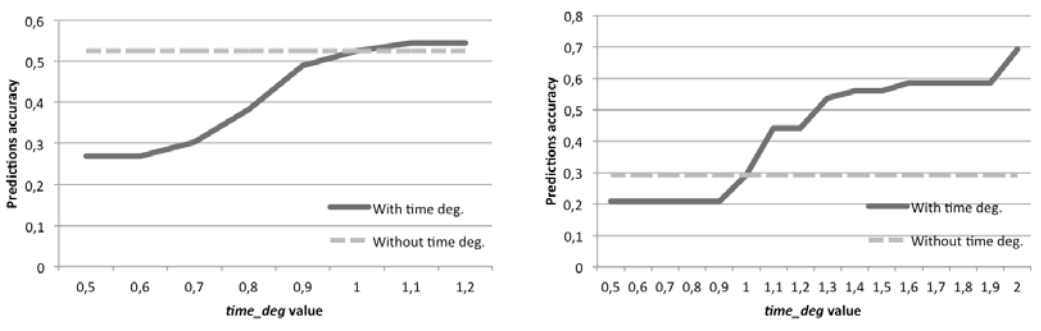


Figure 1. Prediction accuracy for longer anomalies (left) and dramatic change of behavior (right).

The first two situations were evaluated on 8 weeks data and third situation on 10 weeks data. We simulated real usage by iterative addition of the performed pattern and change of the *actual* date. We predicted pattern for the *future* during each iteration and compared our prediction with the pattern that actually happened. The accuracy of a prediction was calculated as a ratio between predicted pattern coverage and actual pattern coverage. If predicted pattern was not *subpattern* of an actual pattern and differed only by one *wrong* location we lower the accuracy by 50 %.

When it had more then one wrong location the accuracy is set to 0. We repeated the evaluation with different positions of the anomalies within the patterns. The final prediction accuracy was calculated as an average of all predictions performed with the same *time_deg* parameter. We can see the results of the second and third situations on Figure 1.

While there is no significant difference in performance with first two model situations, the benefits are clearly visible on third model situation where method with time degradation greatly outperforms the method without time degradation.

5 Conclusion and future work

The main contribution of our work is the method for predicting user patterns in advance. Our method also effectively reflects changes of user's behavior. We experimented with different *time_deg* values and evaluated the adaption rate in different situations. For evaluation of our method with real data we proposed the method for transforming real world GPS coordinates into the patterns, which is based on a combination of several well-known algorithms together with introducing their enhancements. In addition to that we have analyzed several semantics data sources and tested the proposed methods on real world GPS trajectories.

While our method predicts human behavior in advance, combination with existing on the fly approaches would be benefiting, especially when we discover that the user behaves differently that we predicted. Our method is used as a basis for physical activity recommendation. We plan to incorporate also automatic recognition of different means of transportation as it is not always feasible to interrupt the transport. We also continuously work towards improving motivation of users as the motivation greatly influences regular and long-time use of such kind of applications.

Acknowledgement: This work was partially supported by the Slovak Research and Development Agency under the contract No. APVV-0233-10.

References

[1] Ashbrook, D., Starner, T.: Using GPS to learn significant locations and predict movement across multiple users. *Personal and Ubiquitous Computing, 7(5)*, Springer-Verlag, (2003) pp. 275–286.

[2] Assent, I., Krieger, R., Müller, E., Seidl, T.: EDSC: efficient density-based subspace clustering. *In Proc. of the 17th ACM conf. on Information and knowledge management (CIKM '08)*. ACM, New York, NY, USA, (2008), pp. 1093–1102.

[3] Bielik, P., Tomlein, M., Krátky, P., Mitrík, Š., Barla, M., Bieliková, M.: Move2Play: an innovative approach to encouraging people to be more physically active. In *Proc. of the 2nd Int. Health Informatics Symposium (IHI '12).* ACM, New York, USA, (2012) pp. 61–70.

[4] Giannotti, F.,Nanni, M.,Pedreschi, D., Pinelli, F.: Mining sequences with temporal annotations. *In Proc. of the 2006 ACM symposium on Applied computing (SAC '06).* ACM, New York, NY, USA, (2006) pp. 593–597.

[5] Giannotti, F.,Nanni, M.,Pinelli, F., Pedreschi, D.: Trajectory pattern mining. *In Proc. of the 13th ACM SIGKDD international conf. on Knowledge discovery and data mining (KDD '07)*. ACM, New York, NY, USA, (2007) pp. 330–339.

[6] Chon, Y., Talipov, E., Shin, H., Cha, H.: Mobility prediction-based smartphone energy optimization for everyday location monitoring. *In Proc. of the 9th ACM Conf. on Embedded Networked Sensor Systems (SenSys '11)*. ACM, New York, NY, USA, (2011) pp. 82–95.

[7] Pei, J., Han J., Mortazavi-Asl, B., Pinto, H., Chen, Q., Dayal, U., Hsu, M.: PrefixSpan: Mining Sequential Patterns by Prefix-Projected Growth. *In Proc. of the 17th Int. Conf. on Data Engineering.* IEEE CS, Washington, DC, USA, (2011) pp. 215–224.

Article Clustering with Usage of HTML Tags

Peter SLÁDEČEK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
slado89@gmail.com

Abstract. The amount of data on the Internet has been growing rapidly in recent years. This fact has an enormous impact on performance of various systems that people use to find information they are interested in. In this paper we propose a novel approach to data clustering based on analysing the neighbourhood of a web article. The keywords extracted from this neighbourhood are used as input parameters into a weighting algorithm. We think that such modified weighting can give us better overall performance of our clustering algorithm. We present here results of our first experiments that we achieved using this approach.

1 Introduction

People on the Internet are usually flooded with a big amount of web articles while in most cases they search for a specific topic. The searched topics can range from football to economic crisis in the European Union. Authors usually use an interesting heading to make people open and read their article. In most cases these opened articles are not concerned with users' needs. They lose a lot of time during the search, which can be used more effective. On the other side, people ignore not interesting or not-committal name of the article, which can contain a lot of valuable facts for them.

Administrators of online newspapers or magazines are trying to create useful sections for their visitors, which can help them to orientate on their websites. These sections cumulate articles by topic. Nevertheless, this is not suitable for people, who are trying to find a very specific topic.

In this paper we propose a new method for term weighting. We focus only on web articles written in natural language that contain HTML tags. The hyperlink destinations found in analysed documents represent the neighbourhood of an analysed document. We use this neighbourhood for extraction of additional keywords. After that we modify the tf-idf document – term matrix by weights of additional keywords, where the tf component of the tf-idf function is calculated as log(tf). This matrix is an input parameter into a clustering algorithm.

This paper is organized as follows. Section 2 describes related work in the field of term weighting and text clustering. Section 3 introduces our new approach to keyword extraction from the neighbourhood of base document (we explain this concept later). In section 4 we show results of carried experiments and section 5 concludes this paper and proposes future work.

---

* Master degree study programme in field: Software Engineering
Supervisor: Tomáš Kučečka, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 49–54.

2 Related work

Several works focus on keyword weighting. Dumains [3] compares the effectiveness of global weighting at corpuses with various topics. His experiments were carried on the Crangfield corpus, which contains 924 documents on the topic aviation. He identified entropy as the most effective global weighting method. He made a statement that combination entropy with the local logarithmic weighting improves the overall weighting results. Dumains's statement was confirmed by experiments presented in paper [4]. The most clustering algorithms use the tf-idf matrix as an input parameter. Authors confirmed that the tf-idf matrix gives a good result of created clusters, but they also showed that logarithmic entropy can made it better.

In recent years, text clustering has passed a big research. Authors introduced a lot of new methods based on the tree representation. In 2002 the FTC and HFTC algorithm for documents written in natural language were proposed. These two approaches based on the concept of frequent term sets and analysed their behaviour. They used the association rule mining to identity the frequent terms in documents to group them into clusters. In 2003 the research continued with FIHC, which was quicker and more effective that bisecting k-means [1]. FIHC uses word sequences to carry information about word positions.

In paper [5] authors presented CFWS and CFWMS algorithm. The first approach works with frequent word sequences, the second improves words sequences by introducing meaning sequences which are obtained (using WordNet[1]). Experiments showed that the CFWS approach has better performance than FIHC.

In paper [1] authors presented FCDC approach that is based on FIHC. This clustering algorithm works with frequent concepts rather than frequent items. The frequent concept is a set of related words in document, which is used as the measure of documents' closeness. Experiments confirmed that FCDC outperformed all previous algorithms mentioned here.

3 Our method

Text documents usually discuss several topics, which people normally do not notice during their reading. If we want to identify these topics, we need to create an algorithm based on a cognitive process of text understanding or comprehension. Therefore we propose a method that enriches the document text by new keywords extracted from document's neighbourhood. Figure 1 shows the main steps of our algorithm. For now our approach works only with English language.

In the rest of this paper we will use the following terminology, *base document* for the analysed web article and *document* for a web article that comes from the neighbourhood of analysed document.

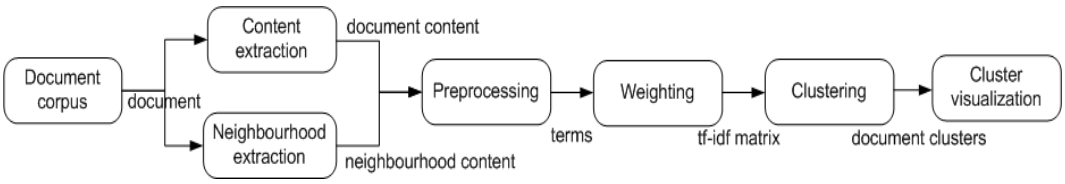


Figure 1. Overview of main steps of proposed algorithm.

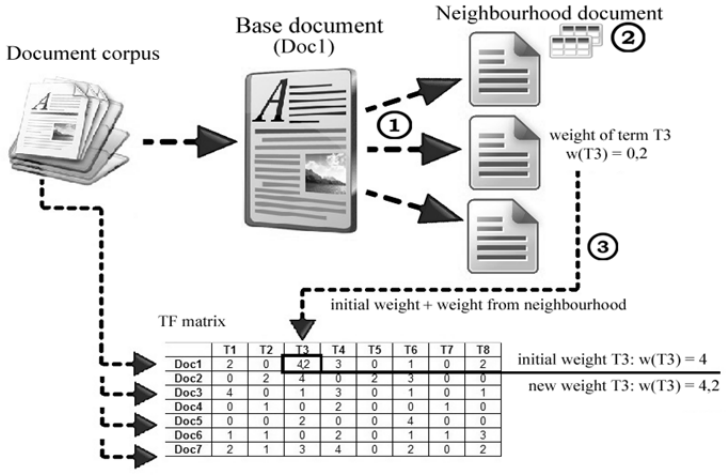
Our approach analyses the base document's neighbourhood for extraction of additional keywords. It uses them as the entry parameter into a weighting process. We divide our approach into the following steps:

1. document neighbourhood acquisition,

[1] http://wordnet.princeton.edu/

2. construction of document vectors,
3. extraction of keywords from document neighbourhood.

These three steps are referred by numbers shown in Figure 2. We will describe them in more detail in the following subsections.



	T1	T2	T3	T4	T5	T6	T7	T8
Doc1	2	0	4,2	3	0	1	0	2
Doc2	0	2	4	0	2	3	0	0
Doc3	4	0	1	3	0	1	0	1
Doc4	0	1	0	2	0	0	1	0
Doc5	0	0	2	0	0	4	0	0
Doc6	1	1	0	2	0	1	1	3
Doc7	2	1	3	4	0	2	0	2

Figure 2. A description of proposed method for extracting relevant information from the neighbourhood of the base document.

3.1 Document neighbourhood acquisition

In [2] authors confirmed that taking keywords from document neighbourhood reflects the topic of an analysed web article (base document). In our case the web article neighbourhood consists of those documents that are referenced through his hyperlinks. A good example of this is Wikipedia[2], where articles commonly contain links on similar topics. Therefore, to extract the base document's neighbourhood we search the HTML code for tag *<a>* and its attribute *href*.

During the extraction process of text content from the identified neighbourhood we need to check, if the text is written in the same language as the text of the base document – English language. For this at least one of the following two criteria must be met:

- website contains hidden information about English language in tag *<html>* and its attribute *lang <html lang="en">*,
- if at least five of the most common words according to [6], occur in the document's text.

In case both of these tests fail, we say that the website's content is too short and it does not matter if its text is written in English or any other language. In such case we are looking for:

- keyword *english* as a text of hyperlink,
- keyword *english* or *en* as a part of link definition (for example as a text of attribute *title* or *href*).

If match is found, we get the URL address and extract its content.

3.2 Construction of document vectors

This is the second step of our approach. After analysing the content of a neighbourhood document, we propose its representation using the two vectors:

[2] http://en.wikipedia.org/

1. *triplet (HTML tag, term, weight)* – triplet represents a term from document neighbourhood,
2. *term vector* – represents all terms, which are extracted of the document neighbourhood.

A process of creating these two vectors (triplet for a term and term vector) is described in the following subsections.

3.2.1 Triplet for a term situated in specific HTML tags

Triplet for a term situated in specific HTML tags is done in the following three steps:

1. Specific HTML tags' text (for instance bold text has bigger importance than normal text) is extracted from the document content and after that saved to the triplet structure (see *Table 1*). If text is composited from more than one word (compound word), we break it up and save each word as an individual triplet.

Table 1. Triplet (HTML tag, term, weight).

HTML tag	word / term	CSS weight of term

2. After creation of all triplets, we check if triplet's HTML tags are not linked to CSS styles. If yes, we need to extract them. Style extraction can be performed in three ways – as element extraction (from attribute style), as extraction from the external place, which is defined with attributes *id* or *class*, or as a combination of those two ways. Computation of CSS weight for term *t* is done by the following equation:

$$cssw(t) = \prod_{i=1}^{n} f(s_i) \tag{1}$$

where $f(s_i)$ is the weight of CSS style s_i. For instance, HTML tag <*p*> has associated two CSS styles – *font-size: large* and *font-size: italic*. Then $n=2$ and total CSS weight of term *t* is calculated by multiplying the weights of these two styles. CSS style weights are from interval (0,2).
3. Triplet for a term contains words, which need to be preprocessed on terms. This preprocessing is executed in the following steps:
   - stop words and number removal,
   - lemmatization,
   - synonym replacement,
   - stemming.

This preprocessing is necessary because in term vector we will use preprocessed terms as a link to the triplet and its information.

3.2.2 Term vector in document neighbourhood

Term vector is constructed in the following two steps:

1. At first, we run defined preprocessing on a natural text of the document content. A result from this process will be saved to the term vector (*Table 2*).

Table 2. Term vector of base document neighbourhood.

term	TF weight	HTML tag weight	total term weight

2. In this step we compute the total term weight using the following equation:

$$htmlw(t) = \prod_{i=1}^{n} f(h_i) * cssw(t) \quad (2)$$

where $f(h_i)$ is the weight of HTML tag h_i and *cssw(t)* is the weight of term *t* based on the extracted CSS style (see equation (1) for more detail). To better explain the equation, we give the following example of an HTML source:

```
<h2 class="aktualita">
     <a href=http://www.fiit.stuba.sk/generate_page.php?page_id=3786>
        IIT.SRC 2013</a>
</h2>
```

In this example we have a heading with a hyperlink IIT.SRC 2013. After preprocessing this text we get terms *iit* and *src*. Because both of them are situated in pair HTML tags <*a*> and <*h2*>, their weight multiplies weight constants of those two tags. These HTML constants are from interval (0,3). If a weight of <*h1*> tag is 2 and a weight of <*a*> tag is 1,5 then the result for terms *iit* and *src* will be 3 (2*1,5). Because HTML tag <*h2*> in this example is linked to CSS style, we need to multiply these values.

3.3 Extraction of keywords from document neighbourhood

Neighbourhood documents are thematically joined with the base document. Because of that we will use term vector to extract some keywords, which help to describe base document. In our approach we choose first ten keywords with the best total weight. This value is depended by neighbourhood document length. After the extraction we insert selected keywords from document neighbourhood into tf-matrix of a base document. If a keyword from this neighbourhood is already present, we modify its weight in the following steps:

- if its weight is equal to zero, we replace it with a new value,
- if its weight is not equal to zero, we sum up a current value with a new value.

This new tf-matrix will be used as an entry parameter to the tf-idf matrix. The tf component of the tf-idf matrix is calculated as log(tf) because it suppresses better terms with high term frequency.

4 Evaluation

We evaluated our approach on a set of 60 articles from BBC Travel[3] written in English language. These articles represented the base documents of our method, each containing 1519 words on average. We compared the results of our experiments with clusters manually created by users. As a clustering algorithm we used FIHC. In carried experiments we decided to focus on the following:

- tf-idf matrix reduction – only nouns are selected from documents,
- tf normalization coefficient – this coefficient is used to reduce the tf weight of keywords extracted from document neighbourhood (at first-time 3 keywords are extracted from each document, the second time 5),
- combination of tf-idf matrix reduction and term frequency normalization – combination of the best solutions from first and second part could gave us better results.

Achieved results are shown in Figure 3. The difference in success rate between document – term matrix containing article words and matrix from only nouns are minimal. As we can see, the matrix created only from nouns has better results than the matrix containing all words. This implies that the preprocessing will take less time if it works only with nouns. In the second part of the experiment, we were looking for a value of tf normalization coefficient, which will maximize

[3] http://www.bbc.com/travel

success rate of the proposed method. Four different experiments have shown that the term frequency of neighbourhood keywords should be normalizing by 0.2 coefficient. This normalization means that all calculated weight of keywords will be reduced to 20 %.

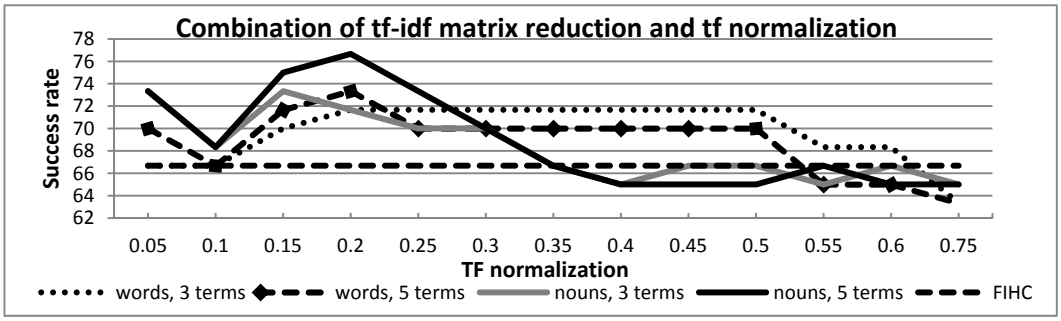


Figure 3. Combination of tf-idf matrix reduction and tf normalization. The horizontal axis represents a coefficient, which normalizes the tf value of neighbourhood keywords. The vertical axis shows success rate of our method.

5 Conclusion

We proposed a new method for clustering text documents based on term weighting. We extended a created term-document matrix with additional keywords extracted from the base document's neighbourhood. We defined two vectors that help us to better represent a text document neighbourhood – a triplet for representing information from HTML tags and a term vector for representing terms with their total weights. Total weight of a keyword is computed from its tf weight, which is multiplied by an HTML tag weight influenced by CSS styles. This means that the weighting process is affected by the HTML and CSS which represent visual importance of term in the web article. This weighting result is an entry parameter to the clustering with FIHC algorithm.

In future work we plan to focus on experiments with weight values of HTML tags and CSS styles. We plan estimate these values by performing experiments on different corpuses. The results should give us sufficient amount of information that we can use to set up optimal values for parameters of our method.

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/0971/11.

References

[1] Baghel, R., Dhir, R.: A Frequent Concepts Based Document Clustering Algorithm. In: *Int. Journal of Computer Applications*, (2010), vol. 4, no. 5, pp. 6–12.

[2] Craswell, N., Hawking, D., Robertson, S.: Effective site finding using link anchor information. In: *Proc. of ACM SIGIR'01*, (2001), pp. 250–257.

[3] Dumains, S. T.: Enhancing Performance in Latent Semantic Indexing (LSI) Retrieval, Bellcore 21236, (1992).

[4] Lan, M., Sung, S. Y., Low, H. et al.: A comparative study on term weighting schemes for text categorization. In: *Proc. of IJCNN'05 on Neural Networks*, (2005), vol.1, pp. 546–551.

[5] Li, Y., Chung, S. M., Holt, J. D.: Text document clustering based on frequent word meaning sequences. In: *Data & Knowledge Engineering*, (2008), vol. 64, issue 1, pp. 381–404.

[6] Oxford dictionary: *The OEC: Facts about the language.* [Online; accessed February 3, 2013]. Available at: http://oxforddictionaries.com/words/the-oec-facts-about-the-language

Relationship Discovery from Educational Content

Petra VRABLECOVÁ*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
petra.vrablecova@gmail.com

Abstract. The domain model is an essential part of adaptive learning system. It expresses the semantics of educational content in the form of metadata. We consider it to be a lightweight ontology, i.e., a set of terms and relations. Manual domain model building is a challenging task for teachers, hence there is an effort to automate it. We propose a method for *automated* acquisition of metadata from educational content, aimed at relationships discovery between terms. We exploit existing methods for relationship discovery from text and adopt them for the educational domain. Our work is promising contribution to the growing field of automated domain model acquisition.

1 Introduction

Abstraction, modularization or building of hierarchies are basic tools for human beings to understand, classify, categorize all sorts of things regardless of complexity. We try to achieve this kind of thinking in machines, too, so the cooperation with them is as meaningful, helpful and efficient for us as possible. To accomplish this behavior we have to supply machines with knowledge humans are able to acquire by modalities and common sense – semantics.

Our work focuses on the area of education, specifically *adaptive learning*. Adaptive learning system stores the semantics of its educational content in a domain model. The domain model is needed as a basis for tracking users' progress in learning and adaption of the content accordingly. It is represented by metadata, in our case a lightweight ontology consisting of set of relevant domain terms (RDT) and relationships between them. Terms represent the semantics of the educational content, which is presented in form of learning objects such as explanations, exercises. E.g., a chapter about file handling would have assigned terms like "write" or "read". RDTs are interconnected by various types of relationships, e.g., is-a, related-to, type-of. Manual creation of the complete and correct domain model is a demanding task for the educational content author (teacher). There are attempts to automate it. Many generic methods for automatic acquisition of metadata have been developed by now. But too few methods focus on area of education. We explored existing approaches, took note of educational content specifics and designed a method for discovery of relationships between RDTs. Our work aims to facilitate the process of domain model acquisition.

---

* Master degree study programme in field: Software Engineering
Supervisor: Dr. Marián Šimko, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 55–60.

2 Related work

There are many generic methods for automatic acquisition of metadata from text. They usually focus only on a part of metadata like terms, concepts, or relationships between them. Natural language processing is widely used in these methods. We focus on methods for relationship discovery. There can be distinguished hierarchical and non-hierarchical relationships. There are two main approaches to relationship discovery – statistical and linguistic.

Statistical methods are based on data mining and machine learning algorithms. Their main advantage is the language independence. On the other hand, to provide good results a big dataset is needed. To discover non-hierarchical relationships the distributional hypothesis (e.g., LSA method) and collocations of words in text are mostly used. Techniques for discovery of hierarchical relationships are usually based on clustering [2], e.g. latent Dirichlet allocation (LDA) [16], formal concept analysis (FCA) [3]. Term subsumption is used to discover hierarchical relationships based on conditional probability of term occurrence in corpus [11].

Linguistic methods are the most often used methods. They depend on the language of the text and require at least basic knowledge about its syntax. But they can provide better results because the discovery rules can be tailored for certain cases of relationship occurrence in text. Techniques based on syntactic dependencies like verb frames [2] and lexical-syntactic patterns [7] or usage of semantic dictionaries like WordNet [10] can be used for both types of relationships.

Only a few works deal with automatic acquisition of metadata for adaptive systems. The authors of MOT adaptive system present method for acquisition of relationships between concepts [4]. Relationship between concepts is created and labeled according to their most common attribute. In [12] is described a method for automatic prerequisite and outcome relationships identification between concepts extracted from a sequentially ordered set of learning objects on C programming language. The disadvantage is the necessity of sequential order of learning objects because it digresses from traditional book or a tree structure of e-courses. An interesting example is the adaptive vocabulary acquisition system ELDIT [1], where methods and techniques of natural language processing were employed in order to create relationships between examples of vocabulary entries and vocabulary entries. The OBAMA-tool [13] aggregates the most of existing freely available tools, techniques and procedures to achieve the semiautomatic building of domain model. The WordNet dictionary is used for relationship identification.

The tool CourseDesigner [14] uses for relationship discovery a vector approach similar to LSA. It is the first tool that also considers the structure of educational content – concepts assigned to learning objects and applies graph algorithms (spreading activation, PageRank algorithm) to improve the results of vector approach. A method for automated hierarchical relationship discovery using the linguistic approach was presented in [15]. This work relies on the specifics of educational content – high occurrence of explanation and determination phrases.

In our work we decided to take advantages of the less explored statistical approach – language independence, no need for syntax knowledge. We assume that educational content has an unambiguously defined narrowed vocabulary which is a precondition for better results of statistical methods. We will make use of the LSA, term subsumption and application of graph algorithms on the educational content's structure to discover relatedness and hierarchical relationships. Our method is described in detail in the next section. We employ the system for educational content management for the evaluation.

3 Relationship discovery

The educational content of adaptive learning system consists of set of learning objects (LO) – any entity, digital or non-digital, that may be used for learning, education or training [9]. We consider mainly text documents. They usually form a hierarchy (a tree or a book structure), i.e., are linked through LO-LO relationships implying their relatedness. The purpose of our method is the auto-

mated creation of a lightweight ontology which will be assigned to the set of LO. A lightweight ontology is considered to be a set of relevant domain terms (RDT) and relationships of different types between them. RDTs are assigned to LOs through RDT-LO relationship that implies the semantic connection between the term and the content of the LO (e.g., the term is a keyword).

Relationship discovery process (see Figure 1) consists of three steps: (a) LO preprocessing, (b) extraction of relevant domain terms, (c) discovery of relationships between terms.

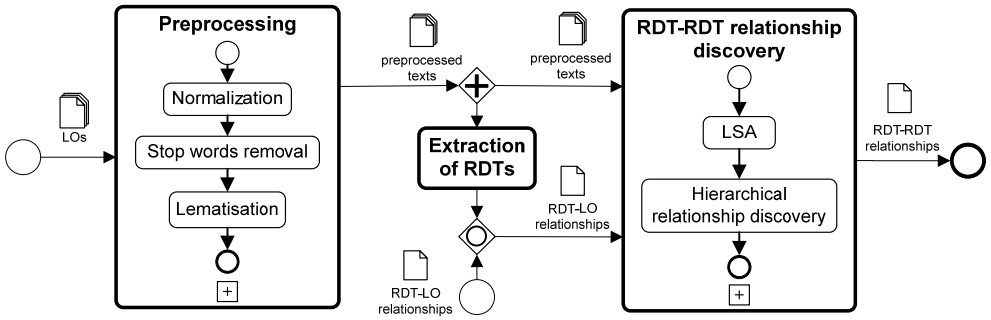


Figure 1. Relationship discovery process.

The input of preprocessing is a set of learning objects. The text of LO is normalized, purged from stop words and lemmatized. The preprocessing procedure depends on the language of the learning objects. Since our relationship discovery method is language independent, in case of other language of learning objects, only the preprocessing procedure needs to be replaced.

Besides preprocessed texts a set of RDTs and RDT-LO relationships are needed. We extract both from preprocessed texts using the standard approach based on tf-idf measure. There can be also used already existing data; the only limitation is the necessity of terms' occurrence in text.

Relationship discovery begins with the construction of a net of RDTs. In this step LSA is applied on the preprocessed texts and assigned RDTs. LSA computes the context similarity of terms, i.e., the similarity of words that surrounds a term in the text. If the similarity is significant, a relationship is created. The output is a set of relationships between related terms. For example in course on programming relationships between terms "function" and "print" or "human" and "user" would appear because these words occur in similar contexts in the learning objects.

LSA is usually used for discovering synonyms in text, therefore we assume that there might exist a hierarchical (is-a) relationship between very related terms. In the next step we propose two variants to determine whether the found relationship is hierarchical.

3.1 Hierarchical relationship discovery based on term subsumption

This variant follows the original work on term subsumption [11] that claims the existence of hierarchical relationship between terms that collocate in documents with the probability of at least 80 %. Then the term which occurs in more documents is labeled as the more general, i.e., superordinate. Since the relationship created by LSA can exist between terms that does not collocate in the same document, in our method we compare sets of learning objects to which are terms assigned (see Figure 2). The procedure for the pair of terms x and y can be described by following steps:

1. Get a set of learning objects with assigned term x.
2. Add the learning objects with assigned terms that are in strong LSA relationship with term x.
3. Construct the set of learning objects from steps 1 and 2 for term y.
4. Compare the sets. If the sets are not disjoint then label the term related to the bigger of sets as superordinate.

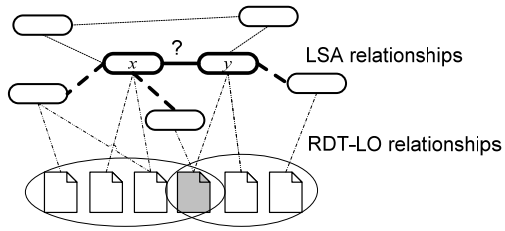


Figure 2. Determination of relationship type between terms x and y.

3.2 Hierarchical relationship discovery using PageRank algorithm

In this variant we apply the PageRank algorithm with priors on the graph of RDT-LO relationships and LO-LO relationships. The set of LO-LO relationships is another input for this method. The procedure of relationship identification between terms *x* and *y* consists of following steps:

1. Apply the PageRank algorithm on the graph with the learning objects assigned to term *x* as the starting nodes for the algorithm and get the sorted list of ranked terms.
2. Repeat the step 1 with the learning objects assigned to term *y* as the starting nodes.
3. Cut the lists' tails and keep only best-ranked terms (first *k* %).
4. Compare the lists. If both *x* and *y* are in both lists and if *x* is on higher position in both lists then label *x* as superordinate.

This technique is based on the Semantic GrowBag algorithm [5].

4 Evaluation

The goal of the evaluation is to find out whether the domain model built by our method is on the level of the manually built domain model. We compare our method result – a lightweight ontology – with the gold standard ontology. To evaluate the hierarchical relationship discovery techniques we compare the results of the algorithms they are based on with our results. The part of the evaluation is also an integration of our method to the system for educational content management.

4.1 Dataset

We perform the tests on the learning objects from the Functional and Logic programming course. The ontology from this course is the gold standard created by the group of domain experts – including the author of the course. The characteristics of the course are shown in Table 1.

Table 1. Functional and Logic programming course characteristics.

	Functional programming	Logic programming
# learning objects	79	42
# words	28,455	23,383
average length of learning object	360.19	556.74
# relevant domain terms	162	138
average relevant domain term length	1.70	1.41
average number of relevant domain terms per learning object	1.94	2.10

4.2 Experiments

In experiments we use the recall and precision measures against the gold standard. We also use these measures for the methods we exploited in our work – term subsumption, Semantic Grow-

back algorithm, to see whether techniques for hierarchical relationship discovery proposed by us give better results. In addition, we modify recall and precision measures with respect to the transitive nature of the hierarchical relationship by following approach in [15].

At the moment we still work on final results. We experiment with various setups of the method and look for the optimal combinations. The best recall and precision are so far 0.59 and 0.08 but we see a scope for further improvement. The preliminary results show that the method has a great potential to supplement methods based solely on linguistic processing (e.g., [15]).

4.3 Integration into COME^{2}T

The part of the evaluation is also the integration of our method into the system COME^{2}T (*COllaboration and MEtadata-oriented COntent Management EnvironmenT*) [6]. Its purpose is the management of the adaptive learning portal's content used to support educational process. This system already contains functionality for non-automated creation of a metadata in a form of lightweight ontology. Integration of our method into the system helps to automate the creation of metadata and ease the work for the authors of content (see Figure 3).

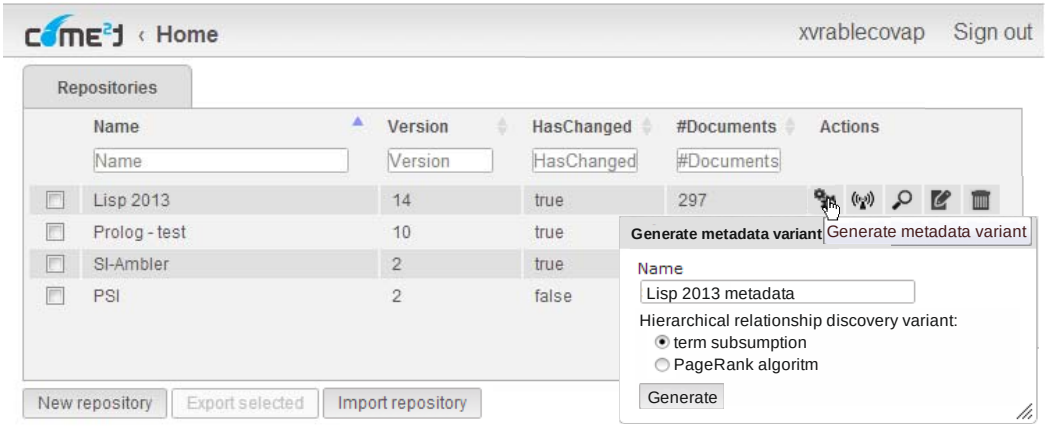


Figure 3. Design of integrated functionality in COME^{2}T system. A repository contains the learning objects from one adaptive educational course.

5 Conclusions

The domain model is important part of the adaptive learning system. It influences the quality of educational content adaptation to the learner. Unfortunately there is not much research on the topic of domain model acquisition and course authoring support. However there are many generic methods for metadata acquisition from text which form a solid basis for research in this area.

In this paper we presented a method for automatic discovery of relationships between terms in a lightweight ontology. We use statistical approach for metadata acquisition from text. The advantage of this approach is its language independency which allows us to apply this method in the future on other than Slovak texts. The uniform vocabulary of educational texts leads to better results of statistical approach. Our method focuses also on the discovery of the hierarchical relationships which comprise the core of the domain model. We took advantage of the specific structure of educational content (hierarchically organized) in this process. The project is at the moment in the phase of evaluation and preliminary results suggest that the most valuable contribution of the method is that it yields different kinds of relationships that cannot be discovered by applying linguistic approaches. As a part of evaluation the method is integrated into the educational content management system to support the course authoring and the automated domain model acquisition.

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/0971/11.

References

[1] Brusilovsky, P., Knapp, J., Gamper, J.: Supporting teachers as content authors in intelligent educational systems. *Int. J. of Knowl. and Learning*, (2006), vol. 2, no. 3/4, pp. 191–215.

[2] Cimiano, P.: *Ontology Learning and Population from Text: Algorithms, Evaluation and Applications*. Springer, (2006).

[3] Cimiano, P., Hotho, A., Staab, S.: Comparing conceptual, divisive and agglomerative clustering for learning taxonomies from text. In: *ECAI 2004: Proc. of the 16th European Conf. on Artificial Intelligence*, IOS Press, (2004), pp. 435–439.

[4] Cristea, A.I., de Mooij, A.: LAOS: Layered WWW AHS Authoring Model and their corresponding Algebraic Operators. In: *WWW 2003: Proc. The 12th Int. World Wide Web Conference*. Alternate Track on Education, Budapest, (2003).

[5] Diederich, J., Balke, W.: The Semantic GrowBag Algorithm: Automatically Deriving Categorization Systems. In: *ECDL 2007: Proc. of the 11th European Conf. on Research and Advanced Technology for Digital Libraries*, LNCS 4675. Springer, Berlin, (2007), pp. 1–13.

[6] Franta, M., Gajdoš, M., Habdák, M. et al.: Management of Lighweight Semantic Content for an Adaptive Web-Based (Learning) Portal. In: *IIT.SRC 2012: Proc. of the 8th Student Research Conference in Informatics and Information Technologies*, Nakladateľstvo STU, (2012), pp. 495–496.

[7] Hearst, M. A.: Automated discovery of WordNet relations. In Fellbaum, Ch., ed.: *WordNet: An Electronic Lexical Database*. The MIT Press, London, (1998), pp. 131–153.

[8] Heyer, G., Läuter, M., Quasthoff, U. et al.: Learning Relations using Collocations. In: *IJCAI 2001: Proc. of the 2nd Workshop on Ontology Learning OL'2001*, (2001).

[9] IEEE LTSC: Draft Standard for Learning Object Metadata. IEEE Standard 1484.12.1. IEEE, (2002), retrieved March 2013.

[10] Lindberg, D. A., Humphreys, B. L., McCray, A. T.: The Unified Medical Language System. *Methods of Information in Medicine*, (1993), vol. 32, no. 4, pp. 281–291.

[11] Sanderson, M., Croft, B.: Deriving concept hierarchies from text. In: *SIGIR 1999: Proc. of the 22nd Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, ACM, (1999), pp. 206–213.

[12] Sosnovsky, S., Brusilovsky, P., Yudelson, M.: Supporting Adaptive Hypermedia Authors with Automated Content Indexing. In: *A3H 2004*: *Proc. of 2nd Int. Workshop on Authoring of Adaptive and Adaptable Educational Hypermedia*, (2004), pp. 380–389.

[13] Šaloun, P., Velart, Z., Klimanek, P.: Semiautomatic domain model building from text-data. In: *SMAP 2011: Proc. of 6th Int. Workshop on Semantic Media Adaptation and Personalization*, IEEE Computer Society, (2011), pp. 15–20.

[14] Šimko, M., Bieliková, M.: Automated Educational Course Metadata Generation Based on Semantics Discovery. In: *EC-TEL 2009: Proc. of 4th European Conf. on Technology Enhanced Learning*, LNCS 5794. Springer, Berlin, (2009), pp. 99–105.

[15] Šimko, M., Bieliková, M.: Discovering Hierarchical Relationships in Educational Content. In: *ICWL 2012: Proc. of 11th Int. Conf. on Web-based Learning*, LNSC 7558. Springer, Berlin, (2012), pp. 132–141.

[16] Yeh, J., Yang, N.: Ontology construction based on latent topic extraction in a digital library. In: *ICADL 2008: Proc. of the 11th Int. Conf. on Asian Digital Libraries*, LNCS 5362. Springer, Berlin, (2008), pp. 93–103.

Asymptotical Sparseness of a Slovak Social Network

David CHALUPA*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
chalupa@fiit.stuba.sk

Abstract. A sparse graph can be informally described as a graph with a relatively low number of edges. We study the asymptotical sparseness as a property of a class of graphs, where the number of edges grows asymptotically slower than the number of pairs of vertices. In such asymptotically sparse graphs, it holds that many algorithms have provably better computational complexity with specific implementation techniques. In this paper, we provide an empirical study of the data from a well-known Slovak social network. We demonstrate that for the class of graphs, obtained by breadth-first search from this network, its metrics indicate the asymptotical sparseness.

1 Introduction

The term sparse graph can be defined in various ways. Generally, a sparse graph is a graph, which has a relatively low number of edges. Therefore, even though the graph can have many vertices, the number of neighbors of a particular vertex tends to be low. In this paper, we study the issue of asymptotical sparseness of graphs. Informally stated, we will call a class of graphs asymptotically sparse if, with growing number of vertices, the graphs of this class remain sparse, even for very large numbers of vertices. This allows us to have more general results on time and space complexity of some algorithms for larger instances of problems.

Intuitively, asymptotical sparseness is an interesting property especially in applications, where the graphs can grow to very large scale, including social networks [12], research citation networks [13], networks of language [2] or the Internet and the World Wide Web. Asymptotical sparseness is a property, which guarantees that an algorithm with linear complexity proportional to the number of edges will be provably better than an algorithm with quadratic complexity proportional to the number of vertices.

In this paper, we present an empirical study of the structure of the data from a Slovak social network, obtained by a web crawler based on breadth-first search. We empirically demonstrate that with growing number of vertices, the number of edges and the change in the degrees of vertices indicate the asymptotical sparseness of the network. Thus, by adding new vertices to the network,

* Doctoral study programme in field: Applied Informatics
Supervisor: Professor Jiří Pospíchal, Institute of Applied Informatics, Faculty of Informatics and Information Technologies STU in Bratislava

it is very unlikely that the sparseness could be violated. This is a practically significant problem, since the asymptotical sparseness has a strong impact on the choice of the right representations and implementation techniques for both exact and heuristic algorithms for problems in such graphs.

2 Asymptotical Sparseness and Its Impact on Graph Algorithms

We begin with the basic terminology, which is related to our topic. Let $G = [V, E]$ be an undirected graph, where V contains the objects (e.g. social network users) and E contains the undirected relations (pairs $\{v, w\}$, where $v, w \in V$). For simplicity, the number of vertices will be denoted by $n = |V|$. The *neighborhood* of a vertex $v \in V$ is defined as $nbhd(v) = \{w \in V \mid \{v, w\} \in E\}$. The number of neighbors of a vertex is called *degree*, i.e. $\deg(v) = |nbhd(v)|$. In this work, the average degree will be denoted as δ and will be studied as a function of the number of vertices, i.e. $\delta = \delta(n)$. The *density* of a graph is defined as the ratio of the number of edges and the number of pairs of vertices, i.e. $d(G) = \frac{|E|}{|V|(|V|-1)/2}$. For an undirected graph, it can be easily shown that $\sum_{v \in V} \deg(v) = 2|E|$ [10].

A common term, which is used to describe graphs with a relatively low number of edges, is that a graph is *sparse*. There are more definitions of sparseness and the choice of the right one depends on a particular application. For our purpose, we consider sparseness as an asymptotical property of a graph. Hence, we define the asymptotical notation in the following way.

For two functions F_n and G_n, we say that $G_n = \mathcal{O}(F_n)$, if the function G_n is upper bounded by cF_n for a positive constant c. The function F_n grows asymptotically slower than G_n, i.e. $F_n \prec G_n$, if and only if $\lim_{n\to\infty} F_n/G_n = 0$ [5].

Now suppose that we move through a graph (in web-based networks, this process is often referred to as *crawling*) and add vertices one by one as they are visited. Then, the number of edges can be perceived as a function of the growing number of vertices, i.e. $|E| = |E(n)|$. For a graph, which is obtained in this way, we say that the graph is *asymptotically sparse*, if the number of edges grows asymptotically slower than the number of pairs of vertices:

$$|E(n)| \prec \binom{n}{2} \equiv \lim_{n\to\infty} \frac{2|E(n)|}{n(n-1)} = 0. \tag{1}$$

Since the average degree δ is defined as $\delta = 2|E(n)|/n$, the condition that $|E(n)| \prec \binom{n}{2}$ can be further reduced to $|\delta(n)| \prec n$, which may be more convenient for practice, as we will show in the empirical part of this work.

2.1 The Impact of Asymptotical Sparseness on Graph Algorithms

A consequence of asymptotical sparseness is that an algorithm, which works in a time proportional to the number of edges, i.e. has an $\mathcal{O}(|E|)$ complexity, is asymptotically faster than an algorithm, which works in a time proportional to the number of pairs of vertices, i.e. has an $\mathcal{O}(|V|^2)$ complexity.

The $\mathcal{O}(|E|)$ complexity is typical for many classical graph algorithms, including breadth-first search and depth-first search [5]. Additionally, many heuristic algorithms have $\mathcal{O}(|E|)$ complexity, including the greedy graph coloring, sometimes also called the Welsh-Powell algorithm [14] and the recently proposed greedy clique covering (GCC) algorithm [3]. These algorithms have the specific feature that they can be used as components of stochastic algorithms, which are repeated many times within a process, which is similar to an evolutionary algorithm. Such an algorithm is called iterated greedy (IG) [3, 6]. Therefore, it is highly relevant that such a component should have as good computational complexity as possible. These greedy algorithms can be trivially implemented to work in $\mathcal{O}(|V|^2)$ time but in asymptotically sparse graphs, it is by far more efficient to use an implementation with $\mathcal{O}(|E|)$ complexity.

In addition, some algorithms are provably faster for asymptotically sparse graphs, when some non-trivial data structure is used. The famous Dijsktra's algorithm with fibonacci heap for the single-source shortest paths problem has $\mathcal{O}(|E| + |V| \log |V|)$ complexity [7], which provably grows asymptotically slower than $\mathcal{O}(|V|^2)$. Therefore, this implementation technique is provably better for asymptotically sparse graphs. Also the Brélaz's graph coloring heuristic, which is very popular in operations research and has many applications [1], can be implemented to run in $\mathcal{O}(|E| \log |V|)$ time with a binary heap. Although this complexity is not provably better than $\mathcal{O}(|V|^2)$ for asymptotically sparse graphs (because of the $\log |V|$ factor), it might still be practically faster than the trivial $\mathcal{O}(|V|^2)$ implementation. We note than $\mathcal{O}(|E|)$ performance can also be achieved in Brélaz's heuristic with a very specific advanced data structure [11].

Last but not least, asymptotical sparseness also has its impact on the cover time of random walks, i.e. the number of steps in a fair random walk on the graph, which are needed to visit each vertex at least once. We note that a random walk on a graph is fair if in each step, each of the vertices to visit is chosen with equal probability. It was previously shown that the expected cover time of random walks is $2|E|(|V| - 1)$ [9]. Therefore, generally, the cover time $C_{|V|}$ of a random walk is $C_{|V|} = \mathcal{O}(|V|^3)$ for an arbitrary graph but for asymptotically sparse graphs, we have a stronger condition that $C_{|V|} \prec |V|^3$.

2.2 Relation to Degree Distribution

An important property of a network is its degree distribution. At this point, we show how it is related to our view of asymptotical sparseness. The degree distribution $P(k)$ is defined as a function, which for each degree k, denotes the fraction of nodes of the network, which have degree k. Therefore, it has similar properties as typical probability distributions.

Let us study the average degree $\delta(n)$ a bit further. Clearly, $\delta(n) = E[P(k)]$, i.e. it is the expected value of the random variable defined by the degree distribution. From the definition of the random variable [5], we have that a connected graph without loops will be asymptotically sparse if and only if:

$$\lim_{n\to\infty} \frac{\delta(n)}{n} = \lim_{n\to\infty} \frac{|E[P(k)]|}{n} = \lim_{n\to\infty} \frac{\sum_{k=1}^{n-1} kP(k)}{n} = 0, \tag{2}$$

where the lowest and highest values of k in the sum are caused by the fact that we have neither isolated vertices nor loops. Thus, in some cases, we might be able to determine the asymptotical behavior of the fraction analytically, thus, proving or disproving the asymptotical sparseness. However, this requires $P(k)$ to be analytically expressed.

It is well-known that social networks are conjectured to be scale-free, which means that $P(k)$ follows a power law, i.e. $P(k) \sim k^{-\gamma}$, where γ is a coefficient of steepness of the distribution [8]. In fact, degree distributions of many real world networks are well approximable by the power law, where the γ coefficicient is usually between 2 and 3. For scale-free networks, we have that the sum $\sum_{k=1}^{n-1} k^{1-\gamma}$ is well approximable by an integral. Hence, the asymptotical sparseness can be decided analytically for ideally scale-free graph classes. In fact, there is a study, which shows that $0 < \gamma \leq 2$ is not likely to occur and for $\gamma > 2$, the networks are relatively sparse [8].

The intriguing part is that the power law still represents just an approximation of the degree distribution. The real-world data might often be quite "noisy". Therefore, the γ coefficient can be fitted by a regression but in fact, we are not sure, whether the distribution really asymptotically converges to $k^{-\gamma}$. Therefore, we have that the analytical approach can be used to prove asymptotical sparseness of an approximation of the data. However, for real-world applications, we might also use a more straightforward approach, as we suggest in the next section.

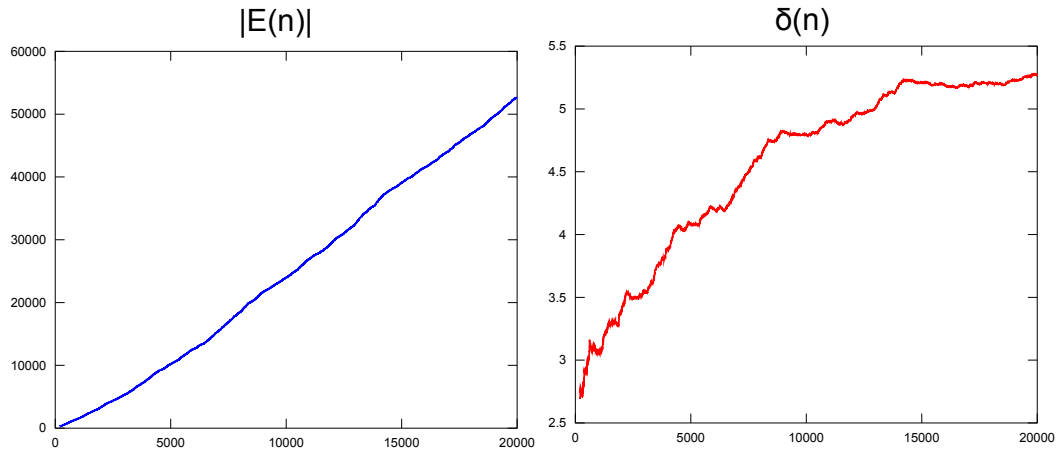


Figure 1. The number of edges $|E|$ and the average degree δ in a growing sample from a Slovak social network.

3 Empirical Study of Asymptotical Sparseness

We now present an empirical study of asymptotical sparseness in the context of one very well-known Slovak social network. For this goal, we implemented a simple crawler, with which we obtained networks of different sizes, with up to 2×10^4 vertices. The crawling algorithm was simply a breadth-first search from a fixed starting vertex of the network.

Figure 1 shows the growth of the number of edges $|E(n)|$ and average degree $\delta(n)$ as functions of n. The number of edges is indeed a non-negative, non-decreasing function. However, it does not yet visually indicate some interesting properties regarding the asymptotical sparseness. Since for an asymptotically sparse graph, it holds that $|E(n)| \prec n^2$, we should decide whether displayed on the left side of Figure 1 is a parabolic curve. From this picture, this is yet not very clear.

However, as we have mentioned earlier, the previous condition for an asymptotically sparse graph can be further reduced to $\delta(n) \prec n$. From the right side of Figure 1, we can see a relatively clear sublinear tendency in the growth of $\delta(n)$, i.e. that $\delta(n) \prec n$, which is an empirical indication of asymptotical sparseness. The function can occasionally decrease, most typically when a new vertex brings only one edge to the network. However, the general trend is a sublinear growth. This means that the new vertices tend to connect to slightly more vertices than the old ones, however, this growth is concave. Thus, the increase in the newly created connections is slowing down with more vertices.

To extend the previous argument, we also calculate the difference of the average degree function. We have taken the change in the values of $\delta(n)$ for steps 1 and 100. The results are displayed in Figure 2. Interestingly, these curves clearly remind one of a derivative of a typical sublinear function, such as $\log n$ or $\sqrt{n}$. We note that the general trend can be seen in only relatively large scale. By "zooming" the graphs, we would not see the change, but especially from the difference function with step 100, it seems that the growth of the function really slows down.

We note that the average degree of a vertex in this network is relatively small due to the fact that the "friendship" paradigm is used very liberally in this web-based social network. In addition, a user has a choice, whether he or she makes his or her list of friends publicly available. Due to this fact, some of the edges of the social network lead to a "dead end". However, this does not affect the representativeness of the network.

Regarding the future work, the degree distribution of this Slovak social network could be studied more carefully. Although this distribution seems to be well approximable by the power law of scale-free networks, it does not follow it entirely and is a bit "noisy" [4]. An alternative approach to

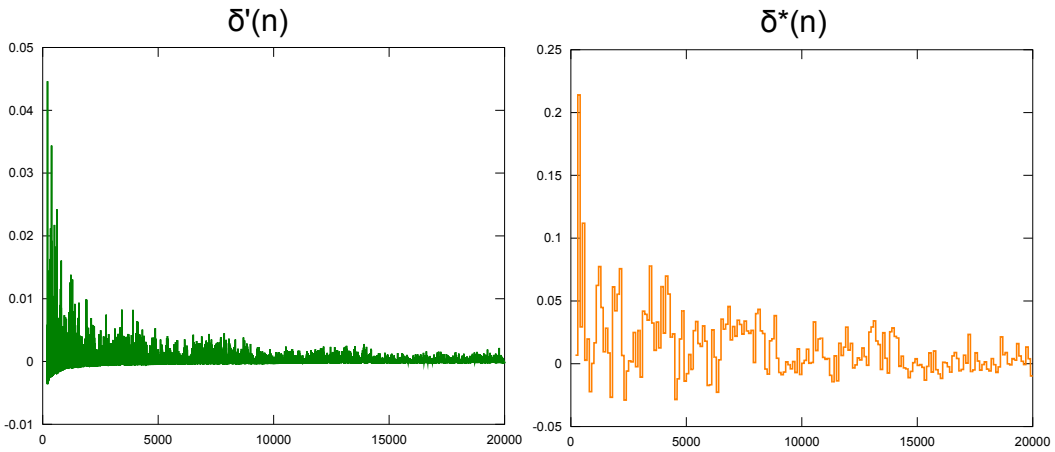


Figure 2. The difference functions δ' and δ^* of the average degree in a growing sample from a Slovak social network, with step 1 *(on the left) and* 100 *(on the right).*

study of the asymptotical sparseness of this network could be that we could assume that the network converges to the power law distribution, approximate the γ coefficient by regression and analytically prove that the approximation of the degree distribution has the property of asymptotical sparseness.

4 Conclusions

In this paper, we empirically studied the asymptotical sparseness of a well-known Slovak social network. By using a web crawler based on breadth-first search on this social network, we obtained a class of graphs, for which we have demonstrated that the average degree seems to grow as a sublinear function of the number of vertices, i.e. $\delta(n) \prec n$. This metric of the network indicates the asymptotical sparseness.

The asymptotical sparseness is somewhat stronger than the simple concept of sparseness of a graph. Asymptotically, the sparseness can be generalized in this way so that also larger samples of the social networks will likely have this property. Therefore, the result presented in this paper, in spite of its simplicity, is significant for further research in heuristic graph algorithms and their hybridizations, by which this study was originally motivated.

Acknowledgement: This contribution was supported by Grant Agency VEGA SR under the grant 1/0553/12.

References

[1] Brélaz, D.: New methods to color vertices of a graph. *Communications of the ACM*, 1979, vol. 22, no. 4, pp. 251–256.

[2] Cancho, R.F., Solé, R.V.: The small world of human language. *Proceedings of the Royal Society B*, 2001, vol. 268, no. 1482, pp. 2261–2265.

[3] Chalupa, D.: On the efficiency of an order-based representation in the clique covering problem. In Moore, J., Soule, T., eds.: *Proceedings of the 14th annual conference on Genetic and evolutionary computation*. GECCO '12, Philadelphia, PA, USA, New York, NY, USA, ACM, 2012, pp. 353–360.

[4] Chalupa, D.: Construction of Near-optimal Vertex Clique Covering for Real-world Networks. Currently unpublished manuscript (available upon request), 2013.

[5] Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: *Introduction to Algorithms (3rd ed.)*. MIT Press, 2009.

[6] Culberson, J.C., Luo, F.: Exploring the k-colorable Landscape with Iterated Greedy. In Johnson, D.S., Trick, M., eds.: *Cliques, Coloring, and Satisfiability: Second DIMACS Implementation Challenge*, American Mathematical Society, 1995, pp. 245–284.

[7] Fredman, M.L., Tarjan, R.E.: Fibonacci heaps and their uses in improved network optimization algorithms. *Journal of the ACM*, 1987, vol. 34, no. 3, pp. 596–615.

[8] Genio, C.I.D., Gross, T., Bassler, K.E.: All Scale-Free Networks Are Sparse. *Physical Review Letters*, 2011, vol. 107, no. 17, pp. 178701–1–178701–4.

[9] Hopcroft, J.E., Karp, R.M.: An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. *SIAM Journal on Computing*, 1973, vol. 2, p. 225–231.

[10] Kvasnička, V., Pospíchal, J.: *Algebra and Discrete Mathematics (Algebra a diskrétna matematika, in Slovak)*. SUT Press, 2008.

[11] Morgenstern, C.: Improved Implementations of Dynamic Sequential Coloring Algorithms. Research Report CoSc-91-4, Texas Christian University, Department of Computer Science, Fort Worth, Texas, USA, 1991.

[12] Pattillo, J., Youssef, N., Butenko, S.: Clique Relaxation Models in Social Network Analysis. In Thai, M.T., Pardalos, P.M., eds.: *Handbook of Optimization in Complex Networks*. Springer, 2012, pp. 143–162.

[13] Sun, J., Xie, Y., Zhang, H., Faloutsos, C.: Less is More: Sparse Graph Mining with Compact Matrix Decomposition. *Statistical Analysis and Data Mining*, 2008, vol. 1, no. 1, pp. 6–22.

[14] Welsh, D.J.A., Powell, M.B.: An upper bound for the chromatic number of a graph and its application to timetabling problems. *The Computer Journal*, 1967, vol. 10, no. 1, pp. 85–86.

User's Satisfaction Modelling in Personalized Recommendations

Michal KOMPAN*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
kompan@fiit.stuba.sk

Abstract. Approaches for the personalized recommendations focus mainly on the user's activity over various portals. User's preferences are not only dependent on the long term history and preferences, but actual user's situation plays crucial role in the user's preferences formation. Thus item liked by user in some context can be disliked in other. Because of this users' variability we propose a novel approach for the user's satisfaction modelling and incorporating the actual user's context and consideration of previous users' rating history. Proposed approach reflects the natural characteristic of user's context, when the various contexts' settings can influence another context. Thanks to our method it is possible increase user satisfaction during one-session recommendation by improving the item's rating predictions.

A paper based in part on this paper was published in Proc. of 8th Int. Workshop on Semantic and Social Media Adaptation and Personalization (SMAP 2013), IEEE Computer Society, pp. 33-38.

* Doctoral degree study programme in field: Software Engineering
Supervisor: Professor Mária Bieliková, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

Attacking the Performance of Okapi BM25 and Tf-Idf

Tomáš KUČEČKA*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
kucecka@fiit.stuba.sk

Abstract. Okapi BM25 is a scoring function that rates a text document based on a given query. It is commonly used as a baseline method in information retrieval. In this paper we propose its usage for keyword extraction together with our own approach that is based on a probabilistic model. We present a novel method that uses the Jensen-Shannon distance to compare keyword distributions over text blocks. Based on the distribution variance the document keywords are divided into global and local. Global keywords characterize the whole document while local only the document's parts. We present our first experiments in which we outperformed the Okapi BM25 and the tf-idf method.

1 Introduction

Keyword extraction process is frequently used in an information retrieval (IR) tasks, for instance to group similar documents together, to provide a set of keywords that best characterize some resource or to cluster text documents based on their topic.

Several approaches to keyword extraction are available and can be based on different models. In this paper we focus on two of them that perform keyword extraction by giving a weight to each term in a given document. One is the well-known *tf-idf* weighting function that is based on a vector space model. The second one, *Okapi BM25* is based on a probabilistic model but it is generally not considered as a keyword extraction approach, although it can be used for term weighting. It is basically used as a ranking function of text documents based on a given query. In this paper we focus on experiments with Okapi BM25 and tf-idf function. We use them to extract keywords from texts and compare the achieved results with the performance of our method.

Our method is based on probability distribution of keywords over text blocks. Based on the occurrence of keywords in these blocks we try to find out the importance of keyword for a document as a whole. For instance, if some keyword occurs only in part of the document it probably would be important only for that part but not for the document as a whole. Although our approach was designed primarily for longer text documents, in this paper we show experiments

* Doctoral degree study programme in field: Software Engineering
Supervisor: Assoc. Professor Daniela Chudá, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 74–81.

carried out on a dataset that contained shorter texts. This is because we were interested how well our approach would deal with such texts.

This paper is organized as follows. In section 2 we introduce the tf-idf and Okapi BM25 functions and give their basic description. Section 3 describes existing work that discusses the Okapi BM25 different versions. In section 4 we describe our own approach to keyword extraction. Section 5 represents the evaluation of the proposed approaches and section 6 concludes this paper and proposes future work.

2 Introduction to Tf-Idf and Okapi BM25

In this section we describe a well-known weighting function tf-idf and Okapi BM25, to which we later refer in this paper. Although the tf-idf approach is generally well known, we give here it's simple explanation.

The tf-idf stands for *term frequency inverse document frequency* and through its *idf* component it introduces a penalization for those words that frequently appear in document's content or in document corpus. It is calculated as

$$tfidf_{td} = tf_{td} \times idf_t \tag{1}$$

where t is a term in document d whose weight is estimated. The tf_{dt} value represents the frequency of term t in document d, which is normalized in order to fit into <0, 1> interval. The idf_t is calculated as

$$idf_t = \log\left(\frac{df_t}{N}\right) \tag{2}$$

where N is the total number of documents in corpus and df_t is document frequency – a number of documents in which term t is found.

The Okapi BM25 is a ranking function that for a given document d and query q containing keywords $q_1, q_2, \ldots, q_T$ calculates the document's score. This score is calculated as

$$okapi_{td} = \sum_{i=1}^{T} oidf_t \cdot \frac{tf_{td} \cdot (k_1 + 1)}{tf_{td} + k_1 \cdot (1 - b + b \cdot \frac{N}{avgdl})} \tag{3}$$

where T is number of all keywords in the given query q and $avgdl$ is the average document length (in words) of all documents in corpus. The k_1 and b are free parameters usually initialized with the following values

$$k \in [1.2, 2.0], b = 0.75 \tag{4}$$

and the $oidf_t$ is calculated as

$$oidf_t = \log \frac{N - df_t + 0.5}{df_t + 0.5} \tag{5}$$

We marked the *idf* component in Okapi BM25 function as *oidf* to distinguish it from the *idf* component in tf-idf function. Robertson and Zaragoza [7] explain the background behind the equations (3) and (5) and the parameter values (4).

3 Related Work

Keyword extraction plays an important part in IR tasks. Many papers discuss this topic and propose various approaches that are, or are not, domain specific. The Okapi BM25 has been a state-of-the-art ranking function for a long time. There exist several variations of this function that are aimed for various domains. In this section we describe some of the existing research dealing with adaptation of Okapi BM25 for different IR tasks.

The fundamental difference between the tf-idf function and the BM25 is that the tf-idf is based on a vector-space model while the Okapi BM25 function is based on a probabilistic model [1, 5]. In this section we will mainly discuss the Okapi BM25 and leave out tf-idf.

In work [4] authors propose a new BM25H ranking functions based on the BM25. The function is used in a web domain to extract keywords from web pages taking into account user's browsing history. The BM25H function watches the freshness of browsed keywords. A standard df_t component is replaced with a new temporal version. Whissell and Clarke [8] proposed a usage of BM25 for text documents clustering. They introduced a *BM25 tf* and *BM25 tf-idf* approach in which the BM25 approach was used as a replacement for the standard *tf* component. The results showed that the modified functions outperformed the original *tf* and *tf-idf* approaches. In work [9] authors introduce a limitation of a classical BM25 when used for very long documents. They introduce an extension BM25L and show that it performs better with no additional computation cost on longer documents. The BM25F presented in [6] is an adapted version of BM25 for ranking structured documents. Authors compared its performance with Lucene's ranking function. Lucene[1] is a text search engine library with advanced features for document processing. The results showed that BM25F outperformed this function in all points especially in case of structured documents. An implementation of BM25 and BM25F is available in current version of Lucene library [3].

4 Our Approach

In this section we present our own approach to keyword extraction which is based on watching distribution of words over text blocks. A text block is basically a block of letters of a firm length.

4.1 Text preprocessing

In the preprocessing process we use standard approaches given in the following list. The order in which these approaches are listed is important as some of these steps cannot precede other.

1. *filtering* – includes removal of all non-letters from document's content and leaving out only one space between every word.
2. *stop-words removal* – stop words are common words that normally have none or very little meaning. They are language specific, for instance in English language these words are *"a"*, *"the"* or *"or"*. In this step we remove all stop words from document's content.
3. *stemming* – is a process of determining stem for every word. We use *Porter stemmer*[2] algorithm to find a word's stem.
4. *block extraction* – the preprocessed text document is divided into blocks of firm length each 130 letters long. This value was estimated empirically based on the carried experiments.

4.2 Keyword extraction process

The keyword extraction process starts after the document's content is chunked on text blocks and important words are extracted from a document. These important words represent keyword candidates and only a keyword candidate can become a keyword.

[1] http://lucene.apache.org/core/
[2] http://tartarus.org/martin/PorterStemmer/

To extract keyword candidates a tf-idf weighting function is used. All words that have a tf-idf value higher than 0.027 are considered important. This value was determined empirically based on experiments. This way we extract hundredths of keyword candidates per document. We can refer to this step as filtration of common words which is the main difference between using the *tf-idf* normally for keyword extraction and, in our case, for extraction of important words.

The next step of the overall process is to determine a distribution vector for every important word. This distribution vector is calculated over text blocks in the following steps.

1. For every important word *w* in document *d* do the following
   a. Calculate word frequencies (tf_{id} values) for every text block in document *d*.
   b. Normalize the calculated frequencies to gain categorical distribution.

Figure 1 shows an example of a word distribution vector that represents a categorical distribution. Categorical distribution represents one trial from multinomial distribution.

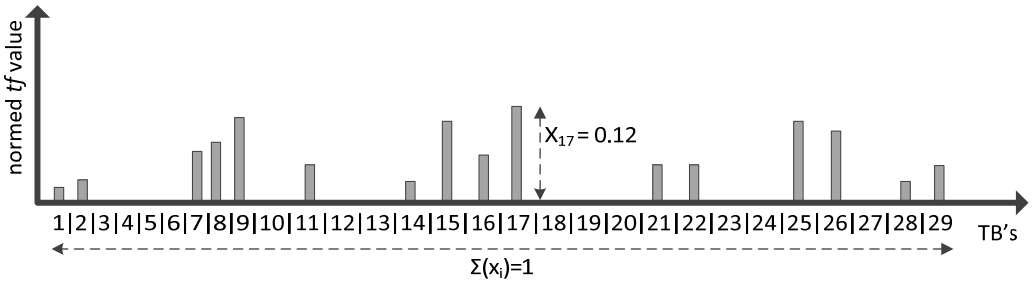


Figure 1. An example of a word distribution vector over text blocks. In the 17th block the normalised tf value of a word is 0.12. We can see that the word is spread across 29 text blocks (x-axis) quite sparsely.

The next step of our algorithm is to find out if a given important word is global or local, i.e. how the word is spread across a document's content. A global keyword should occur more sparsely over text blocks than a local keyword (that is more frequently), because global keyword should be present in all main parts of document's content (introduction, core, conclusion, etc.). Therefore, the variance of a global keyword distribution should be higher than the variance of a local keyword. This is because with raising number of occurrences of keywords in text blocks, the distance from the mean (expected value) raises as well. We calculate the variance for every important word by the following equation:

$$\mathrm{var}(X) = x_i \cdot (1 - x_i) \tag{6}$$

where X is a random variable that represents a realisation from categorical distribution. Then we take top *n* important words with highest variance. Each of these words must occur in at least two different text blocks. All of these *n* selected words are now global keywords. We determined optimal value for $n=8$ which means, that one document can contain max 8 global keywords.

4.2.1 Local keywords extraction

After extraction of global keywords we take the rest of the words and find out which of them are local keywords. Local keyword extraction is performed in the following steps:

1. Calculate distances between all pairs of important words in same document. If the distance between two words is low, these words are called related.
2. Create groups of related words. All important words that at the end of extraction process belong to some group are local keywords.

The main part of this local keyword extraction process is to determine groups of related words. The minimum number of related words in a group is 3. Otherwise a group does not exist. These groups represent local topics. The number of groups in a document can vary. Zero number means that the document discusses no additional topic in its content. When a local keyword extraction finishes, all of the related words from all groups are the local keywords.

To explain the meaning of related words, these are the words that should occur in similar text blocks. For instance, an author normally uses different keywords in a technical part of his work than in the rest of the document. Such keywords are typical only for this part. Therefore, the variance of such keywords' distributions should be low and the distance between their distributions either (because they occur in similar text blocks).

To determine the distance between word distributions we use the Jensen-Shannon (JS) similarity which tells how much work must be done to transfer one distribution into another. The JS distance is calculated by the following equation:

$$JS(P,Q) = \frac{1}{2}\left[KL\left(P, \frac{P+Q}{2} \right) + KL\left(Q, \frac{P+Q}{2} \right) \right] \tag{7}$$

where P and Q are two probability distributions whose distance we are interested in. The KL function is the Kullback-Leibler (KL) distance defined as:

$$KL_d(P,Q) = \sum_{x \in X} P(x) \log \frac{P(x)}{Q(x)} \tag{8}$$

where $P(x)$ and $Q(x)$ must come from a multinomial distribution. This is met as special case of multinomial distribution is categorical distribution. The JS distance is a variation of KL distance that, compared to the KL, is symmetric and limits its output to the <0,1> interval. Both described Dagan et al. [2].

To be able to use the KL distance, we have to satisfy the following conditions:

$$x_i \geq 0, \sum_{i=1}^{p} x_i = 1 \tag{9}$$

$$P(x_i) = 0 \Rightarrow Q(x_i) = 0 \tag{10}$$

where x is a p dimensional distribution vector of word w over text blocks in a given document d. Based on the previous definitions we can see that (9) is fulfilled but (10) not. This is because when comparing two words, normally if one word is not found in a text block A the second word usually is (A represents a concrete text block). Therefore if such situation occurs, we replace the 0 value on the left side with a very small number. This models a situation in which the probability of first keyword appearing in text block A is very small. This allows us to use the KL distance.

4.3 Keyword extraction sum up

Global keywords should reflect the topic of the whole document and according to our approach these are the keywords with highest variance of their distribution vector. Local keywords represent local topics in different document's parts (segments). We use JS distance (7) to determine groups of local keywords. In the next section we focus only on evaluation of global keywords. The local keywords will be a subject of our future experiments and in order to rate the quality of their extraction, we have to perform different experiments from those presented in the following section. Therefore, we do not describe any further details how to detect similarity between documents based on local keywords in this article.

5 Evaluation

To evaluate our approach to global keywords extraction we used a dataset of 220 web articles written in English language from the BBC Travel[3]. All of these articles were manually annotated by people, each assigned with 10 keywords. The average length of an article in the corpus was 559 words. Our aim was to compare the performance of our approach with existing keyword extraction algorithms and to determine optimal values for the following parameters of our method:

- text block length in letters (130),
- *tf-idf* threshold for important words (0.027),
- minimum number of text blocks in which a global keyword must occur (2).

The meaning of these parameters was described in the previous section. The values in the brackets are the optimal values that we determined from the following experiments.

5.1 Okapi BM25 optimal parameter values estimation

Before performing any experiments, we had to determine optimal values for Okapi BM25 parameters *k1* and *b* according to the equation (3). Therefore, we examined the changes in precision and recall for their different values. The main effect on the BM25 performance had parameter *k1* while changing the *b* coefficient had almost no impact on the overall performance. Figure 2 shows the performance of BM25 for different *k1* values.

As can be seen from the figure, with rising values of *k1* both precision and recall increases until certain threshold *t* is reached. After this threshold raising the parameter value has no effect on the overall performance. The optimal value for parameter *b* turned out to be 0 with only little improvement to the performance. The results in Figure 2 were obtained for 0 value of parameter *b*.

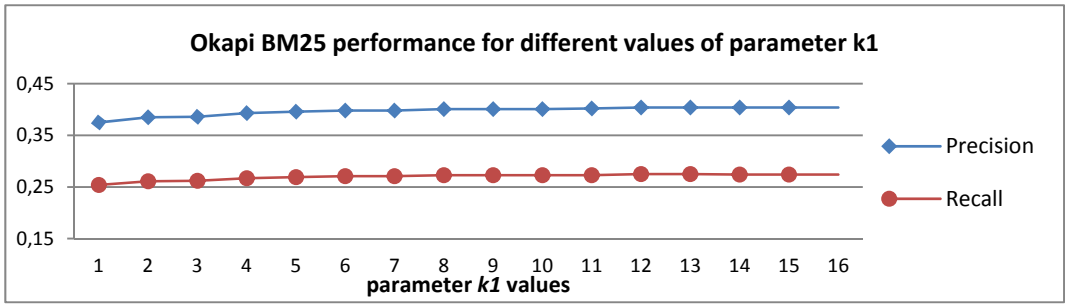


Figure 2. BM25 performance for different values of parameter k1, parameter b = 0.

5.2 Keyword extraction

We used tf-idf, Okapi BM25 and our method to determine keywords for every web article from the BBC travel corpus. As parameter values for BM25, we used *k1=15* and *b=0* (see previous section for explanation). We did not use any background corpus with Okapi BM25 or tf-idf. We watched the precision and recall of the three approaches for different number of keywords at their output. We started the experiments with each approach configured to return 7 keywords and gradually raised this number up to 20. The achieved results are shown in Figure 3 and Figure 4.

In both cases our method outperformed the tf-idf and Okapi BM25. We consider this experiment results as very positive, especially when our method was originally intended for longer documents. The experiments proved that even for shorter texts we are able to determine document keywords better than standard approaches.

[3] http://www.bbc.co.uk/travelnews/

In Figure 3 we can see that our method achieved significantly better precision when more than 12 keywords were extracted for a document (x-axis). This difference in precision is due to several restrictions that our method puts on selection of global keywords. A global keyword must satisfy certain criteria, therefore our method might not be able to extract the required number of global keywords from document's content. For instance, when tf-idf and Okapi BM25 selected 20 keywords for each document, out method selected 16 keywords per document on average. This is because in some documents less than 20 important words fulfilled the criteria for global keywords. Although this behaviour also caused a small drop in the recall measure, our method still performed better than other approaches in both recall and precision measures.

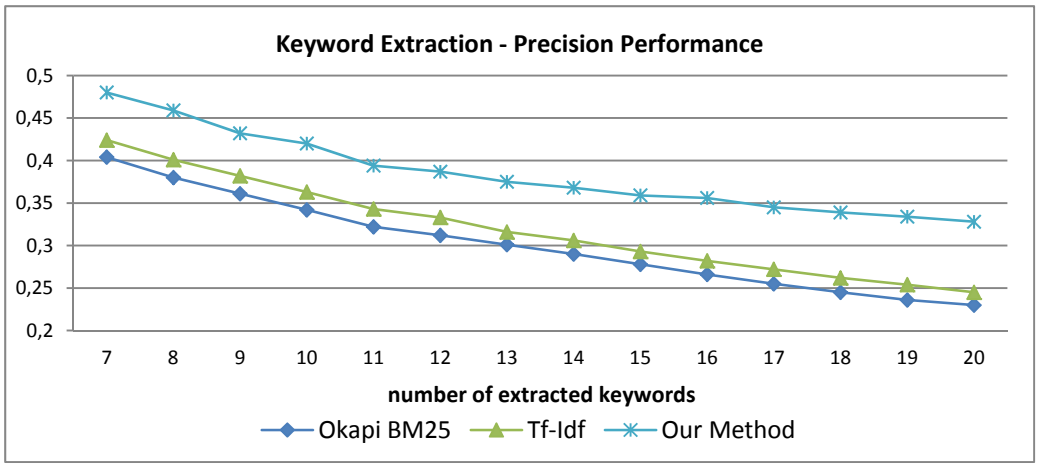


Figure 3. Performance of Okapi BM25, tf-idf and our approach on the keyword extraction problem.

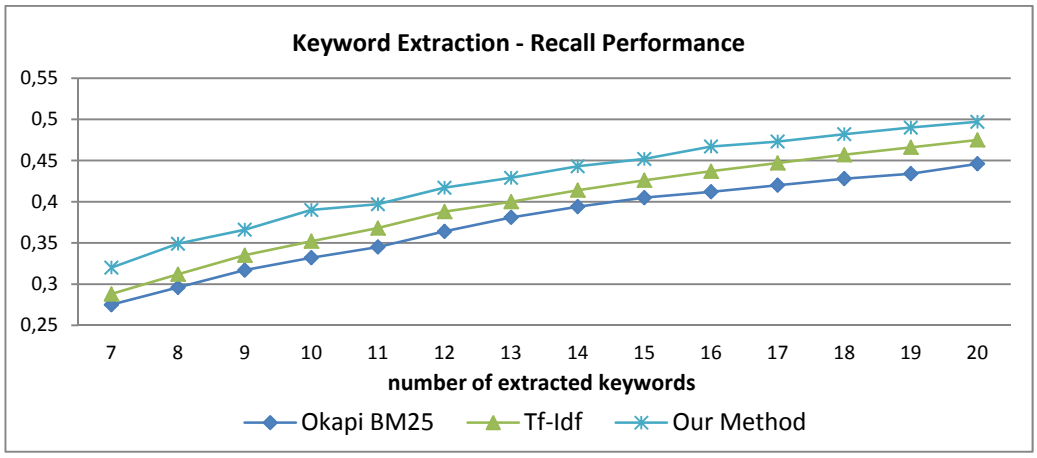


Figure 4. Performance of Okapi BM25, tf-idf and our approach on the keyword extraction problem.

6 Conclusion

In this paper we introduced a state-of-the-art scoring function Okapi BM25 that is widely used in information retrieval to calculate score of a text document for a given query. We mentioned its several variations and used it together with the tf-idf approach and our own method to extract keywords from a text documents written in natural language.

The main contribution of this paper is in proposal of a novel approach to keyword extraction based on comparing word distributions over text blocks in document's content. The experiments showed that our approach outperformed both the tf-idf and Okapi BM25 in the quality of extracted global keywords. We achieved noticeably better performance in both precision and recall measures. On average we had +0.06 precision and +0.03 recall when compared to the second best approach. Nevertheless, additional experiments have to be performed to confirm these results.

An interesting outcome of carried experiments was the performance of the Okapi BM25 method which was slightly inferior to the tf-idf approach. We explain this by its complexity. Although BM25 is a state-of-the-art function we presume that it must be adapted through other parameters for keyword extraction. For instance the Lucene implementation of BM25 for keyword ranking contains additional normimg coefficient.

6.1 Future work

Our next steps will be to implement other variations of Okapi BM25 function and use them for keyword extraction. We plan to perform additional experiments on larger datasets containing more documents and longer texts. We are especially interested in performance of our method on longer texts where we expect even better results.

Yet in this paper we did not mention any experiments with local keywords that together with global keywords represent output of our method. In this document we only explained how we plan to compare the local keyword distributions – extract local keywords. Therefore, in near future we also plan to focus on experiments with local keywords.

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/0971/11.

References

[1] Bennett, G., Scholer, F., Uitdenbogerd, A.: A comparative study of probabilistic and language models for information retrieval. In: *Proc. of the nineteenth conference on Australasian database (ADC '08)*, Australian Computer Society, (2008), vol. 75, pp. 65–74.

[2] Dagan, I., Lee, L., Pereira, N.C.F.: Similarity-Based Models of Word Cooccurrence Probabilities. *Machine Learning – Special issue on natural language learning,* (1999), vol. 34, no. 1–3, pp. 43–69.

[3] Joaquín, I.P., et al.: *Integrating the Probabilistic Models BM25/BM25F into Lucene*. [Online; accessed February 19, 2013]. Available at: http://arxiv.org/abs/0911.5046

[4] Karkali, M., Plachouras, V., Stefanatos, C., Vazirgiannis, M.: Keeping keywords fresh: A BM25 variation for personalized keyword extraction. In: *Proc. of the 2nd Temporal Web Analytics Workshop (TempWeb '12)*, ACM Press, (2012), pp. 17–24.

[5] Manning, D.C., Raghavan, P., Schütze, H.: An Introduction to Information Retrieval. Cambridge University Press, Cambridge, England, (2009).

[6] Pérez-Agüera, R.J. et al.: Using BM25F for semantic search. In: *Proc. of the 3rd International Semantic Search Workshop (SEMSEARCH '10)*, ACM Press, (2010), pp. 1–8.

[7] Robertson, S., Zaragoza, H.: The Probabilistic Relevance Framework: BM25 and Beyond. Foundations and Trends in Information Retrieval, (2009), vol. 3, no. 4, pp. 333–389.

[8] Whissell, S.J., Clarke, L.C.: Improving document clustering using Okapi BM25 feature weighting. *Information Retrieval* (2011), vol. 14, no. 5, pp. 466–487.

[9] Yuanhua, L., ChengXiang, Z.: When documents are very long, BM25 fails!. In: *Proc. of the 34th international ACM SIGIR conference on Research and development in Information Retrieval (SIGIR '11)*, ACM Press, (2011), pp. 1103–1104.

Usability of Anchoring Algorithms for Source Code

Karol RÁSTOČNÝ*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
rastocny@fiit.stuba.sk

Abstract. Metadata of dynamic data, e.g. source code, have to be maintained after each modification of describing data. The first step of metadata maintenance is a repair of metadata anchoring after a modification. Many anchoring approaches exist for regular texts or web pages, but they are not directly applicable for a source code. We propose information tags' anchoring for source code. Our proposal contains a definition of the robust location descriptor and the algorithm for building and interpreting the descriptor. We evaluate our approach on the dataset based on change-sets from commercial projects that contains more than sixty thousand C# files.

A paper based in part on this paper was published in 24th International Conference on Database and Expert Systems Applications (DEXA 2013), Springer, pp. 372-379.

* Doctoral degree study programme in field: Software Engineering
Supervisor: Professor Mária Bieliková, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

Beyond Code Review: Detecting Errors via Context of Code Creation

Dušan ZELENÍK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
zelenik@fiit.stuba.sk

Abstract. Our intention is to support the process of code reviewing. The idea is in detection of possible mistakes which could be created during software development. Unlike syntactical analyses or detecting smells we focus on the human factors. We presume that developer is affected by variety of conditions which could be present during the code creation. To detect conditions with negative impact on the code quality we observe developer and his history in software development process. Using source control management and bug reports we discover relations among mistakes and conditions. For instance, working too long could be one reason to make a mistake in software development. We used logs of real software developers and their activities collected in almost one year. We analyze source control management, activities on PC and actions in IDE. We also experiment with bug reports in offline evaluation of our method and its precision of error detection.

1 Introduction

Every human has his own patterns in behavior. Everything we do has its reason. We react to different situations, different states. Therefore has the environment, in which we exist, huge impact on our outputs. When we talk about outputs, we have to mention our productivity and efficiency in work. It is definitely affected by our current situation and surroundings. Every human who is trying to accomplish something has to be focused on the task. Usually only experts repeating the routines do not need to focus so much that they are able to work in any state of their environment as it was discussed by Milton [7].

In our work we focus on software developers and their productivity and effectiveness which is influenced by surroundings. Programmers do lot of mistakes when they are not in the shape. We are going to prove this assumption by analyzing their behavior while developing software. In our work we focus on two main aspects which affect the quality of programmer's outputs.

– *Continuity of work.* This means that programmer is working in continuous time and he is not interrupted by external happening. Interruptions could cause problems with refocusing

* Doctoral study programme in field: Software Engineering
Supervisor: Professor Mária Bieliková, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 90–97.

on the task and reconstructing the situation. This leads to loosing time and loosing context and eventually leads to mistakes and incomplete tasks.

– *Stereotyped work.* When programmer works in common situations or in the common environment, he is used to conditions what positively affects his outputs. This also means that programmer need some sort of stereotype in work. Loosing the stereotype brings anomalies in his behavior that cause anomalies in his outputs. Anomalies in outputs could emerge into mistakes.

Measuring the quality of programmer's outputs is not trivial issue. There are many different metrics which we could used. However most of them are not very precise and mostly incomparable among different programmers. Simple metric would be calculating the number of lines of operations in code which was created during specific amount of time. However, this only calculates the quantity but not quality of outputs. We assume that we should use two metrics.

– *Commits.* Commit is an action of programmer which is done regularly to submit some part of the work. Commits submitted into source control system are treated as logical end of programmers effort. Commits are usually executed when something is accomplished. We consider number of commit as metric which shows the success.

– *Bugs.* Even if programmer has done something, we could cause some mistake which has, however, been revealed later. We are counting commits which are fixing bug associated with previous commits. Such a bugfix commits are negative metric which enables us to express low quality of programmer's work.

In modern IT world we wrongly focus only on analyzing the code. Code analyzing is on the other hand very successful but we identified, that we should analyze programmer himself to recognize all aspects which could cause mistakes in code. Because commonly used automated tools for detecting mistakes in code could not be 100% reliable and we still can not recognize all the mistakes done by programmers.

Code as an output of programmer needs to be checked because mistakes which could occur. In the past we needed to manually check every operation before the code was executed. Preparing the punched card for program execution was more rigid since there was no way back than remaking the card. Every mistake was punished by repunching process. However, nowadays we use automated tools such as syntactical analyzer and debuggers. Or even more advanced code smell detectors. Furthermore, our code is often generated automatically so we know that it is going to be correct. These automatic tools help us in the process known as code review.

We are going to support this process by identifying part of codes which were created in bad conditions and are probable to contain some mistakes. This is new approach which is focused on analyzing programmer instead of code he produced. Our approach lays in determining the context of programmer which could be associated with lower quality of his outputs. We analyze programmers activity by tracking his behavior. These analyses are then used to detect parts of code which were created in this context.

There many types of bugs which could be present in code. Most of the syntactical error are revealed immediately. There are also many tools for identifying code smells. But there is almost no chance to identify logical mistakes or mistakes associated with business goals of the software. It only means that there are still some mistakes which we had to reveal in code review or testing. Our approach faces this problem and our effort is to create assistant to recognize mistakes during code review or testing.

As for experimentation we decided to focus on analyzing programmers' activities. We show some experiments with detecting *continuity in programming* and *stereotyped programming*. Our intention is to present that productivity and efficiency of programmers are influenced by these two assumptions.

2 Related Work

In the work by Czerwinski et al. [2] authors analyzed behavior of workers. The nature of tasks of these workers was rather parallel. Multitasking suggests that these workers cannot work continuously on single task. They had to switch among tasks. This leads to frequent interruptions. The point of their work is to show how workers react to these interruptions. They want to design software for task management which is fond of bad habits caused by interruptions.

Workers observed in this study are programmers. They work in `Matlab` but their tasks are usually fragmented due to secondary activities such as reading emails or preparing presentations. The study also showed that only 18% are dedicated to projects and productive tasks. The rest of the activities are secondary. 7% of total tasks are those which were interrupted by other tasks. 40% of tasks were self initiated, what means that user was not interrupted by external influence but on his own. Rest of the interruptions were external.

They also observed that tasks which were interrupted were twice as long than tasks of the same nature that were not interrupted. This was mostly caused by reconstructing the context after interruptions (searching for previously edited files). All of these findings are later followed by proposal of a software which helps to reconstruct the lost context after interruptions.

Another work made by authors Mark et al. [6] is done by observing 24 information workers. Authors approached these workers via task fragmentation. They considered two components of work fragmentation. Length of the task and number of interruptions. They empirically proved that 57% of tasks are interrupted generally. They also proved that workers who are collocated are working longer and rate of their task interruptions is higher (every 3 minutes in top).

They divided tasks into primary (full responsibility – 87% of tasks) and secondary (part of a team – 13% of tasks) and observed the rate of fragmentation. In comparison, primary tasks had around 60% of fragmentation and secondary had almost 42% fragmentation rate. Interesting is, that primary tasks were mostly interrupted by metawork and secondary tasks were mostly interrupted by personal reasons.

Another finding is, that longer the worker is working on a task, higher is his tendency to be interrupted. Worker seems to be more open to interruptions in case he is working too long on the task. Their ultimate intention was to design support tool for keeping continuity in task accomplishing.

Work by Parnin and Rugaber [9] was dedicated to similar analyzes. They observed 86 programmers and 10 thousands sessions. This research is mostly on analyzing the reconstruction of context after interruption. This is typically represented as time needed to get back to previous task. This lags also exists before programmers starts to work on the task but they usually observed it after interruptions. They also observed the navigation and patterns of navigation to reconstruct previous state.

There were more strategies which programmers usually use to get back to interrupted task. These strategies were not exclusive. Programmers usually returns to last edited method (almost 15% of sessions) with the lag mostly less than 1 minute, searching for context by navigating (56%), executing the program (59%), reading notes (43%), reviewing task description (75% of sessions and causing 30 minutes lag in average), reviewing code history (59% of sessions and causing more than 15 minutes lag in average).

Another paper by Parnin [8] is on using mental state of programmer to recognize the intensity of their focus on work. They presumed that a programmer's mental state when he is deeply focused could be observed by movements of tongue, mimic muscles, lips and vocals. This is another way to gather information on programmer's state and determining quality of his work. They used EMG to detect any changes during programmers work.

In the work by Iqbal et al. [5] they observed workers with camera. They tracked pupil movements. They compared self-initiated interruptions in easy tasks and difficult tasks. This revealed that workers are tend to interrupted difficult tasks very often.

In the later work by Iqbal et al. [4] they tried to train a method for automated interruption detection. They used previous videos and participants to create a dataset to reveal when user was

interrupted. Further distractions are then automatically revealed by analysing video of worker solving task.

Work by Trafton et al. [10] is facing the problems which lead to lowering the quality of programmer's outputs. They try to prepare user and help him to get back to the state which was more suitable for work. This is done by identifying upcoming loosing of optimal state and preparing user environment to get back to this state.

Fogarty et al. [3] tried to automatically differentiate between productive and non-productive situations of the worker. The goal was to recognize when the worker could be interrupted and when it is not suitable. It could be used to reduce unwanted interruptions or simply to notice user that he is in the state which is easily reconstructed for further work on the task.

3 Tracking the Developer

Programmers usually do many actions while programming. We can track almost all of them by monitoring their PCs [1]. In our work we only focus on activities reletad to work on PC. We also could observe their faces and their surrounding environment. We also could track their phones and similar devices which could influence their state and quality of outputs. However, action on computer are satisfying for the research which we do.

We track sessions which start when programmer logs into the computer and ends when he is logged off. Each session contains events which are related to computer and running applications. Since we work with programmers we observe mostly development platforms. We track every applications and events such as changing focus etc. We also track more particular events associated with IDE they use. Typical session of one programmer could be demonstrated as it is in Figure 1. This figure shows that user changes his state from non-productive to productive. However, we could discuss what was productive and what was not. Browsing for materials on the website could help to solve some issues related to code. In our work, we consider such a behavior to be non-productive, because it is lowering the focus of the programmer. We also claim that too much events in IDE also suggests that programmer is not productive. He is trying to apply one of the strategies to reconstruct the context and work on the t ask. The most productive time in the illustrated session is when user is actively in the IDE and he does not frequently invoke events in this IDE. He is actually creating something new. He occasionally saves, builds or tests his work. He does no switching or searching in files.

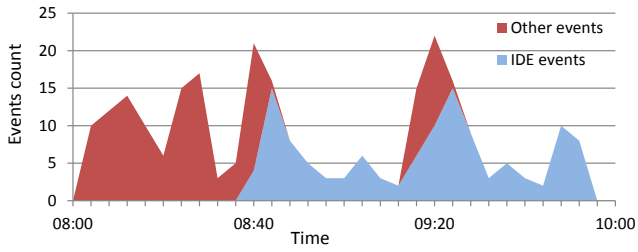


Figure 1. Events during session. We can observe more event types which occurred during programmers work. Those peeks in IDE events actually means that programmers tries to reconstruct previous work session.

We also track commit of the programmer. We presume that every programmer is doing commits when he successfully finishes some logical part of the task. We track these events. Every commit contains information on the files which have been changed. We also track special types of commits which are fixes for bugs. These commits could be recognized using keyword `bugfix` in their description. These commits are retrospectively assigned to previous commit and we mark work which has been done before this commit as ineffective. Commits actually indicate the incremental

work of programmer which is productive and could be effective in case these commits did not contain errors.

Using these observation we identify which states of the programmer are productive and which are effective. Productive state means that user was really working on something. Effective state means that he was productive and created correct output. State diagram of the programmer could be summed up in the Figure 2. We enriched this diagram with inheritance to simplify transitions (e.g. productive is the superclass of effective and ineffective).

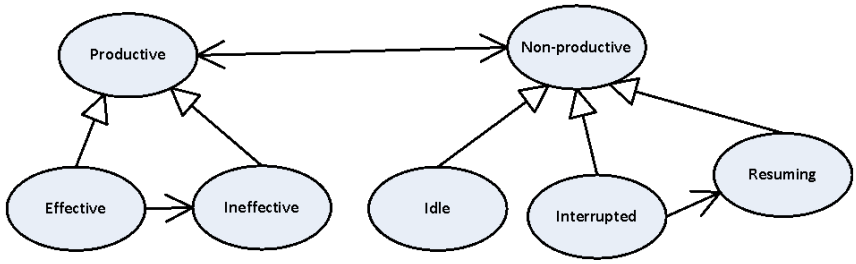


Figure 2. States of the programmer. We are interested in effective, ineffective, productive and non-productive state.

4 Method for Identifying Situation Related Code Errors

We mostly discussed interruption in programmer's work. These interruptions are actually reason why programmer looses the optimal state for working on tasks. Loosing the concentration is one reason to do mistakes. We also claim that programmer could create more mistakes while in bad state caused by inappropriate conditions. For instance, he is working too late in the night, or he is working very long.

Our method for identifying these situations is based on detecting interruptions and anomalies in programmer's coding manners. When we know when the programmer was interrupted and when he was working in inappropriate states we mark code which was produced. We mark this code with probability calculating while discovering interruptions and anomalies.

4.1 Discovering interruptions

Interruptions is easily recognized. We can see that programmer was working in IDE and his work was interrupted by different events. Everything what happens in the state when he reconstructs previous state and could be understand as problematic. In other words, the rate of interruption frequency of one committed work is the probability of leaving mistakes. We calculate the number of interruptions during a session and calculate the probability of error occurrence within commit.

```
commits.each do |commit|
  commit.continuity = 0
  fragmentation = 0
  commit.sessions.each do |session|
    fragmentation = (session.hours / 2) / session.interruptions.length
    commit.fragmentation += (fragmentation / commit.sessions.length)
  end
end
```

Notice that we used presumption that 2 hours session could be interrupted once. In other words we presumed that there the longest session lasting exactly 2 hours. We used these presumption due to experiment made in previous work by Parnin et el. [9] where authors claimed that this is the longest session captured during their observation.

4.2 Discovering anomalies

Anomalies are not so easily discovered. We need to know what are the routines of programmer. We use these routines to filter out each event which happened in common conditions. Those which left are probably causing some mistakes.

```
conditions = {}
commits.each do |commit|
  commit.sessions.each do |session|
    session.states.each do |state|
      if (state.focus == 'IDE')
        conditions[state.conditions] ||= []
        conditions[state.conditions] << state
      end
    end
  end
end
routines = conditions.sort_by { |key, value| value.length }\
routines.each do |routine|
  routine.states.each do |state|
    conditions[state.conditions].frequency = routine.length / routines.last
  end
end
```

And finally, to calculate the error rate we calculate the rate of anomalies in one commit. Commit is then marked with probability of error occurrence. We compute the rarity of actions made before commit. Rarity actually means, that we use conditions which were actual during task completing. This could be any set of conditions. However, in our work we only used time in the meaning of rareness (day of a week, hour of a day).

```
commits.each do |commit|
  commit.sessions.each do |session|
    rarity = 0
    session.states.each do |state|
      rarity += (conditions[state.conditions].frequency / states.length)
    end
    commit.rarity += (rarity / commit.sessions.length)
  end
end
```

4.3 Probability of error in commit

We showed how to compute the fragmentation of work before commit and rarity of actions which have been made before commit. These two values from interval $< 0,1 >$ need to be merged to express the probability of error in commit. In other words, more rare are the actions before commit and more fragmented is the work before commit, bigger is the chance to make a mistake.

$$error\_rate = 2 * \frac{rarity * fragmentation}{rarity + fragmentation} \tag{1}$$

Table 1. Experiment with two metrics and revealing code errors. We used threshold for probability of error occurrence stated to 10%.

metric	totally revealed	correctly revealed
work fragmentation	10	7
anomalies in work	3	2
metric combination	6	6

5 Experiments and Evaluation

We explained our approach to calculate error rate in finished work. We took two different metrics to calculate this rate. Fragmentation of work and its rarity are two attributes which we focused for examination. To prove our idea that these two metrics are influencing the error rate in code we need to know if there was really a mistake in the code which we marked as possibly wrong.

Our intention is to use special commits which were described as bugix commits. Since every commit is associated with some files we can say which commit was fixed. Normal commit changes a set of files. Bugfix commit changes another set of files. These two commits are related through intersection of this files. However, this is not the best way to reveal real errors, but we have no better information on bugs.

Another problem is that bugfix commits are fixing only those errors which were already discovered. There could be mistakes which are not fatal. These mistakes are usually more difficult to discover by human since they are not so obvious. These mistakes could be related to design patterns, smells etc. Everything is working fine with these mistakes but they are causing problems with further coding or understandability of the code. So this eventually means that there is no sense for measuring false positive error discovery.

There is also no sense for computing false negatives, since we do not claim to reveal all mistakes which could be revealed by other methods. These methods were actually used to discover errors in the dataset we used. This explains why we only used true positives to calculate our precision.

As for dataset, we used 5 programmers and they work tracked during less than one year. Our dataset contains mainly events which happened during tracking. We have all events on computer such as changing application, running application. We also have events focused on IDE. We record event such as creating files, projects, saving files, deleting files, copy-pasting among windows etc. We only need to differentiate whether or not event happened in IDE.

Applying our method for discovering errors in code we present a Table 1.

6 Conclusions

In our work we focused on two metrics which could influence the quality of task accomplishing. We worked with 5 programmers who have been tracked during the period shorter than one year. We used these metrics to reveal mistakes in code. However our dataset contains just few programmers and we are able to positively reveal only few errors with relatively high threshold of 10%.

Anyway, our research showed some promising results. We based our effort on previous work in this field and their empirically confirmed presumptions. We hope that by enlarging the dataset by more programmers and more events we could outperform our current results.

Acknowledgement: This contribution is the partial result of the Research & Development Operational Programme for the project Research of methods for acquisition, analysis and personalized conveying of information and knowledge, ITMS 26240220039, co-funded by the ERDF.

References

[1] Bieliková, M., Návrat, P., Chudá, D., Polášek, I., Barla, M., Tvarožek, J., Tvarožek, M.: Webification of Software Development: General Outline and the Case of Enterprise Application Development. Procedia Technology. *Procedia Technology*, (to appear).

[2] Czerwinski, M., Horvitz, E., Wilhite, S.: A diary study of task switching and interruptions. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '04, New York, NY, USA, ACM, 2004, pp. 175–182.

[3] Fogarty, J., Ko, A.J., Aung, H.H., Golden, E., Tang, K.P., Hudson, S.E.: Examining task engagement in sensor-based statistical models of human interruptibility. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '05, New York, NY, USA, ACM, 2005, pp. 331–340.

[4] Iqbal, S.T., Bailey, B.P.: Understanding and developing models for detecting and differentiating breakpoints during interactive tasks. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '07, New York, NY, USA, ACM, 2007, pp. 697–706.

[5] Iqbal, S.T., Zheng, X.S., Bailey, B.P.: Task-evoked pupillary response to mental workload in human-computer interaction. In: *CHI '04 Extended Abstracts on Human Factors in Computing Systems*. CHI EA '04, New York, NY, USA, ACM, 2004, pp. 1477–1480.

[6] Mark, G., Gonzalez, V.M., Harris, J.: No task left behind?: examining the nature of fragmented work. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '05, New York, NY, USA, ACM, 2005, pp. 321–330.

[7] Milton, J., Solodkin, A., Hluštík, P., Small, S.L.: The mind of expert motor performance is cool and focused. *NeuroImage*, 2007, vol. 35, no. 2, pp. 804 – 813.

[8] Parnin, C.: Subvocalization - Toward Hearing the Inner Thoughts of Developers. *2011 IEEE 19th International Conference on Program Comprehension*, 2011, pp. 197–200.

[9] Parnin, C., Rugaber, S.: Resumption strategies for interrupted programming tasks. *Software Quality Journal*, 2010, vol. 19, no. 1, pp. 5–34.

[10] Trafton, J., Altmann, E.M., Brock, D.P., Mintz, F.E.: Preparing to resume an interrupted task: effects of prospective goal encoding and retrospective rehearsal. *International Journal of Human-Computer Studies*, 2003, vol. 58, no. 5, pp. 583–603.

Web Science and Engineering

Extracting Interesting Information from Social Media

Tomáš JÁNOŠÍK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
janosik.tj@gmail.com

Abstract. Social network is phenomenon, which according to its popularity and interesting concepts, gives us a lot of impulses to study it. In this paper we discuss the problem, which interesting information can be found in social network and how to extract this information from the messages of the social network users. Because our aim was to gain interesting information, we focused on extracting the user's opinions and emotions, which they express in their messages and statuses, and studied with their help the field of recommending the news from news agencies.

1 Introduction

Nowadays, the amount of information surrounding us became so huge that one cannot comprehend everything and know every news that appear on the Internet. Informatization, globalization and the widespread technologies, allowing people to connect through social media, enable us to get the news from anywhere in the world. We were monitoring Sandy in USA, nuclear tests in North Korea or The Arabic Spring. We could see the role that social media played mainly during the events of The Arabic Spring. Social media gave the revolutionary ideas the boost to spread and connect people in Arabic world to form a mass.

It is known that when the hottest topics appeared online on the web of news agencies, was this topic already discussed on the social media. The social media thus give us some kind of time advantage as we observe and search for the top news, but news agencies have the advantage of complexity, order and organisation of the offered information.

As we mentioned above, the huge amount of new information is the obstruction in human effort to keep in touch with new information. Through human physical and mental limitations we cannot read each new article on the Internet. There is the need for recommender system enabling us to select and filter information with high value for us. Personalization is important aspect of studying the recommender systems. It is our goal to provide sufficient information personalized for the user. Another important factors of adaptation and success of recommender systems are trend-awareness and local-awareness as we can see from [6].

---

* Bachelor degree study programme in field: Informatics
Supervisor: Professor Pavol Návrat, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 101–106.

In this paper we propose new method for recommending interesting information. We utilize opinions and critiques freely available on social media. People spontaneously need to talk about interesting breaking news and many of them use social media for this purpose – to be the first to come in with something new and to share our opinions of recent events with our friends. Therefore, if we were able to detect such opinions in social media content, we could recommend news appropriately. Opinions and emotions could have various forms, thus we employ two approaches to analyze social media content.

First of them is able to capture human emotions. Only the interesting news could make us react emotionally, therefore emotions in messages (concerning news) may be the sign of something interesting or really breaking. The latter approach is able to capture opinions in a complex way. Considering news recommendations, we are interested in messages containing judgements, appreciation, positive and negative opinions and this approach uses knowledge from psychology and lexicology to analyze text and capture opinions. Both of them will be later discussed in detail.

2 Related work

As part of our research, we studied several recommender systems. Every system has its advantages and also limitations. Basic recommendation offer webs of the news agencies. Along with the articles can be found recommended top articles possibly divided into several categories, e.g. top read, top discussed or top articles chosen by the editor. Some of these recommended articles may be misleading, because user do not search for the top read article, but for the article that interest him.

There are two main groups of recommender systems: *collaborative* and *content-based* systems. Google personalized news [4] is example of the collaborative system. This system takes its advantage from huge amount of users. Taking the click on the news link from user as he likes it, can be this system effective at collecting very popular news. Personalization then utilizes reading history of each user to give appropriate recommendation, because if many users with reading history similar to particular user like some article, that user probably would like it too. This system can process huge amount of information, but lacks considering negative user opinion.

2.1 Recommender systems and social media

Using social media in recommender systems is not rare. Chen et al. [3] propose system which recommends interesting URL links to the user. This system utilizes *bag-of-words* to represent user profile as well as profile of the URL link. Recommendation is made subsequently comparing these two profiles. In their work authors propose the concept of serendipity. While some users have exact interest and do not have time for other areas of interest, some may be looking for unusual and unknown. This concept can be used to solve the problem of new user, while having not so much information about him, providing him some unusual recommendations that could help him to develop his interests.

Twitcident [1] is the name of another content-based recommender system. It was created to help emergency services getting the further information about incidents in real world. People close to particular incidents share on Twitter their observations and this system offers such information to the emergency services. To search for relevant information on Twitter, this system utilizes *Named Entity Recognition* (NER) to recognize relevant tweets from irrelevant. It utilizes structured data available on the Internet, e.g. DBpedia or Alchemy. Ability to offer most recent and relevant information is the main advantage of this system.

One of the most recent approach that utilizes information available on the Twitter to recommend news was proposed by De Francis Morales[5]. This system extracts named entities from news articles, from tweets from particular user and from tweets from his friends. Friends on Twitter are users that are followed by the particular user. Relevance of tweets from user and from

his friends to particular news article can be measured considering extracted named entities. Most popular entities discussed by the user and his friends are used to recommend personalized news. Important factor for this system to recommend recent news is time. By observations, published news are considered uninteresting after two days.

Another recent approach proposed by Phelan [7] combines the RSS feeds and tweets from Twitter to recommend news personalized to specific user. The user needs to define interesting RSS feeds, and these feeds are then used as source of information for recommendations. At the core of this system, most discussed terms (on Twitter and RSS feeds) are collected. Using TFIDF method, the terms are then used to evaluate score of each news article. Basically frequency of occurred terms is crucial for this system and there may be some interesting articles, which could stay dissuaded, because of high occurrence of targeted terms.

To this time, many news recommender systems were created, but few of them are used widely. The recall of recommended information increases with every new approach, although we cannot judge which one gives the best recommendations. Still, there is a gap to be filled to provide more accurate recommendations with low error rate. In our opinion, we can extract better information from social media through emotional text analysis. With the comprehension of human emotions and opinions, we are able to provide to our users news more accurate to their interests.

3 From news to recommended news

The description of individual parts of the recommender system is provided below. The parts of the system and relations between them are shown in Figure 1.

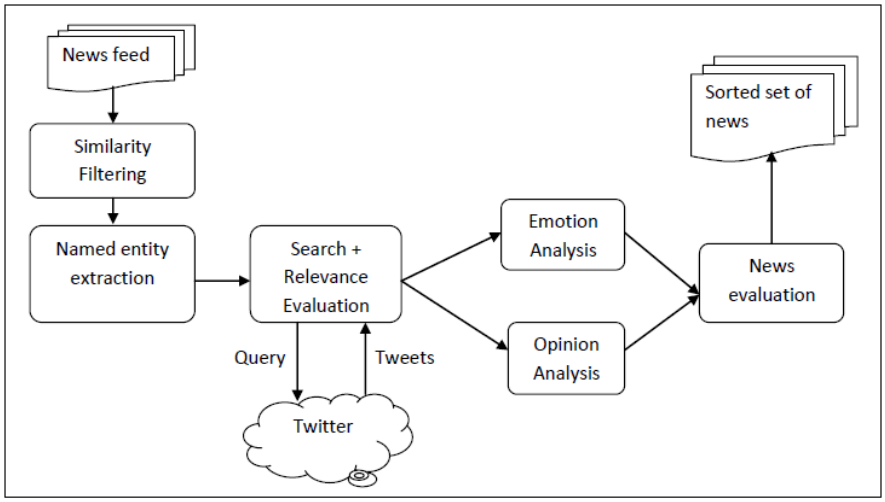


Figure 1. Parts of the proposed system and the basic flow.

The main source of news for our system is BBC World. News from their RSS feed are the basic input to our system. It is unnecessary to filter this stream, because some news are updated through the time. It is simple to represent news story. We utilized NER, which have been proposed in several recommender systems and tested as suitable. DBpedia offer several ways to recognize named entity and huge amount of structured text[1]. Representation of news as a set of named entities give us the advantage to comprehend the semantics of news story.

[1] We have discarded entities of type "Thing", because the entities not in DBpedia database are marked as "Thing".

Named entities are suitable for searching for the tweets, because they are one of the basic methods to express topic in the tweet. The form of such expression may vary, because of special characters specific for Twitter, e.g. '#', but used Twitter's Search API sees no difference between the various forms.

Social media are unlimited source of information. They provide us the information about social connections of users, groups of people related to users and also information about their opinions and emotions. The freedom of publishing any information and the accessibility of such publishing enables users to express themselves.

There are several ways to analyze the tweets. We utilize two approaches that have their basis in psychology. Both analyze text published by the users, but each of them provides better understanding of either emotions and opinions. Both uses corpuses, in which are words represented as a combination of features. In following sections we will discuss them in detail.

3.1 Emotion analysis

This approach was proposed by Akcora [2] to discover breakpoints in public opinion. Every word can be express as vector of eight types of emotions:

{Anger, Sadness, Love, Fear, Disgust, Shame, Joy, Surprise}

Every word can express zero or more of these emotions. The words that do not belong to any type of emotions are neutral words. Neutral words cannot help us to discover interesting information from social media. Emotion value of each word is number of types of emotions it express. To measure emotion value of some tweet, we have to measure emotion value of each word. Emotion value of the tweet is then sum of the emotion values of the words in tweet.

This analysis filters interesting tweets containing additional information about news. There is no difference between tweets expressing positive or negative emotions, because both of them carry additional value for recommendation. Although, the emotions are important to detect interesting tweets, to filter out the tweets that do not contain any opinion, but emotions, we utilize Opinion analysis.

3.2 Opinion analysis

This approach was proposed by Whitelaw [8] to measure attitude value of film reviews. It is based on appraisal groups. There are four main groups which can further divide. Basic division and possible values are shown in Figure 2.

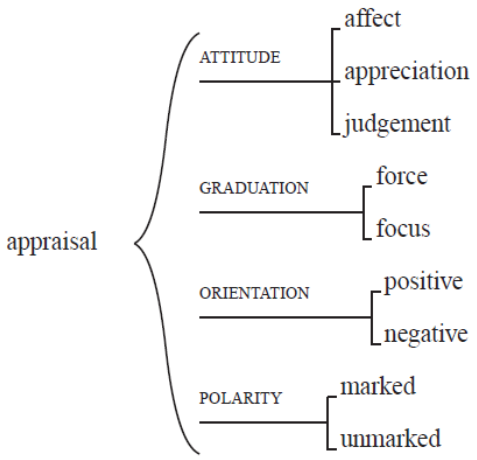


Figure 2. Basic division of words based on appraisal theory.

Not all words have their appraisal value. We have built our appraisal corpus, where the words have their *Attitude*, *Graduation*, *Orientation* and *Polarity*. This system goes behind the words. In our corpus, there are larger lexical units, which consist of more words. For example the phrase "not very happy" together has different attributes as "happy". The word "not" affects the Orientation value as well as Polarity. Opinion value of the tweet is then computed as sum of occurrence of units of appraisal corpus in the tweet.

3.3 News evaluation

Emotion and opinion analysis have the complementary function. The system mixes these approaches of text analysis to sort the news from news feed. We utilize the advantages of each of these approaches to get more appropriate recommendations. The score of each news is then shown in Equation 1:

$$V_N = \sum_{\forall T \in N} aE_T + bO_T \tag{1}$$

where V_N is total value of the news, E_T emotional value, O_T opinion value of the tweet T relevant to news N and the values are normalized by the coefficients a and b.

The system sorts the news set by the value obtained by News evaluation and presents to user top k interesting news.

4 Evaluation

In our work we had to evaluate three methods:

- getting the relevant tweets for particular news,
- emotion and opinion analysis,
- satisfaction and recall of our system.

To evaluate relevant tweets for particular news, we have made many observations. From these observations we have discovered, that the queries for relevant tweets give the best results, if the query consist of two named entities of different types. Also the less have the entities in common, the better is the relevance of queries. For example if one of named entities was some country and the other entity was city in this country, the relevance of returned tweets went low. The most relevant tweets the Search API returned in case of named entities of type person.

To the time of writing of this article, we have evaluated only emotion analysis. Over 80 % of relevant tweets were considered as neutral (emotion value) and having low value (opinion value), but the rest we evaluated as relevant and suitable for recommendations. We have to evaluate opinion analysis now and inspect the influence of various combination of parameters a and b to score and recall of news evaluation.

For further research we have goals to utilize emotion and opinion analysis for recommending interesting news to the user and to try to measure the utility of this approach and satisfaction of the users.

5 Conclusions

In this paper we presented the state-of-the-art in the area of news recommender systems. We discussed some advantages and limitations of these systems and proposed a new approach to news recommendation. Using messages on the social media is suitable and tested approach as information base, although we had to tackle some challenges specific to our field of use. Messages on the social media are unlimited source of various kind of information. We had to use

unfortunately limited computational and networking technologies, what we had to consider in the process of development.

We proved that emotion and opinion analysis could lead to increased performance of recommender systems, although some aspects of the use remain untested. There are numerous useless messages on the social media and we will make an effort to be more precise in filtering useful information. The perspective outlook motivate us to further research and to test our system in real use.

Acknowledgement: This work was partially supported by the Slovak Research and Development Agency under the contract No. APVV-0208-10.

References

[1] Abel, F. et al.: Semantics + Filtering + Search = Twitcident Exploring Information in Social Web Streams. In: *HT '12: Proceedings of the 23rd ACM conference on Hypertext and social media*, (2012), pp. 285–294.

[2] Akcora, C. G. et al.: Identifying Breakpoints in Public Opinion. In: *SOMA '10: Proceedings of the First Workshop on Social Media Analytic*, (2010), pp. 62–66.

[3] Chen, J. et al.: Short and Tweet: Experiments on Recommending Content from Information Streams. In: *CHI '10: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, (2010), pp. 1185–1194.

[4] Das, A. et al.: Google News Personalization: Scalable Online Collaborative Filtering. In: *WWW '07: Proceedings of the 16th international conference on World Wide Web*, (2007), pp. 271–280.

[5] De Francisci Morales, G. et al.: From Chatter to Headlines: Harnessing the Real-Time Web for Personalized News Recommendation. In: *WSDM '12: Proceedings of the fifth ACM international conference on Web search and data mining*, (2012), pp. 153–162.

[6] Kanta, M., Simko, M., Bielikova, M.: Trend-Aware User Modeling with Location-Aware Trends on Twitter. In: *SMAP '12: Seventh International Workshop on Semantic and Social Media Adaptation and Personalization.* IEEE, (2012), pp. 23–28.

[7] Phelan, O. et al.: Using Twitter to Recommend Real-Time Topical News. In: *RecSys'09: Proceedings of the Third ACM Conference on Recommender Systems*, (2009), pp. 385–388.

[8] Whitelaw, C. et al.: Using Appraisal Groups for Sentiment Analysis. In: *CIKM '05: Proceedings of the 14th ACM international conference on Information and knowledge management*, (2005), pp. 625–631.

Trending Words in Navigation History for Term Cloud-Based Navigation

Samuel MOLNÁR*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
molnar.samuel@gmail.com

Abstract. Tag clouds provide an alternative and more readable navigation interface by exploiting visual features of words placed in a cloud and augmenting their information value with different font size and color. We propose a method for term cloud navigation which exploits navigation history as a source of metadata for personalized navigation. We consider a position of a word in a query as important and rank the list of the documents accordingly. We also introduce trending words in users' navigation history as a relevant factor determining users' interests while navigating. We performed an evaluation of our method in Annota, bookmarking and annotation system.

A paper based in part on this paper was published in Proc. of 8th Int. Workshop on Semantic and Social Media Adaptation and Personalization (SMAP 2013), IEEE Computer Society, pp. 53-58.

* Bachelor degree study programme in field: Informatics
Supervisor: Professor Mária Bieliková, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

Changes of User Interests in Time and Their Application in Search Engines

Roman BILEVIC*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
roman.bilevic@gmail.com

Abstract. Human interests are not stable and search engines should take this information into consideration. The goal of this paper is to develop a method of user interests modeling that captures the influence of time. This method uses keywords to represent the user interests. The algorithm that is used to create model itself is known as divisive hierarchical clustering (DHC).The information about changes in user interests are captured when the user is browsing on the Web in form of implicit feedback. The modified three-descriptor representation is used to express the time influence on user interests. The created model is used to improve the search.

A paper based in part on this paper was published in Cognitive Traveling in Digital Space of the Web and Digital Libraries (Tradice 2013), STU Press, pp. 50-55.

* Master degree study programme in field: Information Systems
Supervisor: Tomáš Kramár, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

User Interest Modelling Based on Microblog Data

Miroslav BIMBO*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
bimbo08@student.fiit.stuba.sk

Abstract. User model is a digital representation of a real user, necessary for providing personalized behaviour in information systems. Creating user model based on microblog data has a great potential due to the amount and nature of data produced by users. On the other hand, it is a nontrivial problem due to the shortness of microblog posts and specific language used in them. It is necessary to cope with these problems by linking posts to external sources by using various matching criteria. It is easier to extract more user related information from "richer" and well-structured external sources, than from original post. We proposed a method for user interest modelling, utilizing several enrichment methods and aggregating their output together.

1 Introduction

Microblogging services give users opportunity to publish 140 characters long posts reflecting arbitrary aspect of their life to the Web. These kinds of services become very popular last years – for example, Twitter had up to 140 million users, producing 340 million posts every day in 2012[1]. This massive amount of content created by users, describing their everyday opinions, emotions or interests, became soon a very interesting data mining source for researchers.

Some useful information can be extracted from microblog data, e.g., actual trends, news, product ratings, or even earthquake reports [8]. This paper is focused on extracting information about users who are writing the posts, i.e., creating the user model. User model can be used with advantage to recommend or filter content for a particular user, helping him to overcome information overload and allowing him to focus attention on relevant information resources.

On the other hand, leveraging microblog as a data source is a nontrivial problem. Specific language and shortness of posts are the characteristics, which make the processing of microblog data a challenging task. The state-of-art approaches often do not perform very well, therefore it is important to seek for some new solutions, modifications or combinations of existing approaches.

In this paper, we present a novel method for user modelling based on microblog data. Related works (Section 2) are followed by proposed method (Section 3), its evaluation (Section 4) and conclusions (Section 5).

---

* Master degree study programme in field: Information Systems
Supervisor: Dr. Marián Šimko, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

[1] http://mashable.com/2012/03/21/twitter-has-140-million-users/

2 Related Work

According to what is being modelled, we can distinguish several basic views on a user model. Most often, the six user characteristics are modelled [9]: Demographic information, Interests, Goals and Tasks, Knowledge, Emotional State and Context. The characteristic most relevant to our work is user interest – the second most used characteristic on the web [7].

User interests can be represented in straightforward way as a vector of *keywords* [4]. The problem of such representation is ambiguity, heterogeneity and low ability of generalization. The same problems as the keywords have also the *hashtags* [1]. Another representation of user interests are *latent topics* [6, 10, 11], which address the most of problems related with keywords. On the other hand, identification of topic for the particular microblog post using this representation has half precision compared to random keyword from post [3]. Using the vector of *Semantic web entities* [1, 5] has shown good results in filtering or recommending content.

In many works the interests of particular user are extracted only from posts written by given user [4, 10]. Interesting way of improving the user model is microblog post enrichment. Enrichment methods extract user interests from documents, which are only related to original user post. In some works, interests are extracted from microblog posts of users, who are related to original user (e.g., followees). Such methods can achieve better results – if interests are chosen as 20% of related users' interests and 80% of original user interests [6].

Based on assumption that microblog posts are often related to news, Abel et al. proposed method, where user interests are extracted from news articles [1]. They found the weighted relations between microblog posts and news articles based on their similarity and temporal properties. If the weight of relation is high enough, user interests for a given microblog post are extracted from a given news article. The method enables to create fuller and richer user model. Bernstein et al. proposed method for extracting topics of interests leveraging Yahoo search [3]. It is based on transformation of post to query and extraction of topics from the given search engine result. Using this method, they achieved better results compared to extracting the topic from a post itself.

In our work we build the user model based on similar approach as [1, 3]. However, rather than focusing on improvement of one enrichment method, we research a proper combination of more methods and aggregation of their results.

3 Enrichment Methods Combination and Aggregation

We propose the method for user interest model creation. It is based on the intuition, that we can build better user model by aggregating results of several different enrichment methods. The model is represented as vector of pairs – interests (semantic web entities) and their relevance to the user.

We investigate in importance of weights given by enrichment methods, by semantic web entities extractor and by a classifier which compute, how much are particular posts interest-related. Furthermore, we propose methods aggregating same interests found more times in one post, or aggregating same interests found in more posts.

The process of building the user model is following:

1. Classification: Compute, how much the post is interest-related (compute interest relevance *i*).
2. Enrichment: Find relations between enriching documents and post by each particular enrichment method (compute confidence of relation *c*).
3. Interest extraction: Extract interests (represented as semantic web entities) from given resources using the OpenCalais web service (web service returns weight *w*).
4. Weighting: Compute importance of each particular interest for a user (compute *score*).
5. Aggregation: Aggregate same interests through particular methods and posts (compute *aggregated score*).
6. Filtration: Filter out low score interests and mostly repeated repeating false positive interests.

3.1 Microblog post classification

Interest relevance *i* is a weight computed by a classifier, which we train using a supervised machine learning algorithm. We create train set for classifier by manual annotating of posts with interest relevance, getting the pairs: <post, interest relevance of post>.

The following features of posts is used to train a classifier: is retweet, is reply, sentiment, contain mention, start with mention, length, contain opinion, contain signs, contain uppercase word, contain slang, contain shortcuts, time, probability of posting in given time, non-dictionary to dictionary word ratio.

3.2 Post Enrichment and Interest Extraction

To enrich microblog posts and extract interests from posts we employ methods presented in Table 1. Division of methods to *internal* and *external* relates to the fact, if the enriching document is microblog post or not. Method relation type is *implicit*, if relation between enriching document and post can be found directly in post content, and *explicit*, if the relation is derived or induced.

Table 1. Proposed interest extraction methods and their characteristics.

Method name	Description	Source type	Relation type
Baseline	Interest extraction from text of given post.	Internal	Explicit
Hashtag	Interest extraction from microblog posts, which contain same hashtag, as given post.	Internal	Explicit
Tagdef	Interests extraction from descriptions of hashtag (from Tagdef service, http://tagdef.com)	External	Explicit
URL	Extract interests from text of URL included in given post	External	Explicit
News	Interest extraction from text of most similar news article (same as used in [1], entity based method).	External	Implicit
Youtube	Firstly transform the microblog post to query (same as used in [6]), than extract interests from descriptions of first N returned videos.	External	Implicit

We employ the described methods to extract user interest. For each extracted interest are assigned the following weights:

- weight [*w*] given by semantic web entity extraction service (how much the service trust, that extracted entity is correct),
- confidence [*c*] given by enrichment method (how much the method trust, that associated enriching document is correct, e.g. similarity between news article and post),
- interest relevance [*i*] given by interest classifier (how much we believe, that the source post is interest-related).

For each user interest extracted by a method from a particular post, the score is computed as:

$$score(user, interest, post, method) = w^{\lambda_1} * c^{\lambda_2} * i^{\lambda_3}, \lambda_i \in \{0,1\} \quad (1)$$

Our goal is to find the appropriate coefficients (i.e., find which weights are significant), which result in the most accurate user model.

3.3 Interest Aggregation

The goal of interest aggregation is to compute the *aggregated score* of given interest for a user:

$$aggscore\ (user, interest) = AF_p(\ AF_m(\ score(user, interest, post, method)\ )), \quad (2)$$

where AF_m is aggregation function through all employed methods and AF_p is aggregation function through all user posts.

AF_m has to cope with the situation, when one interest is found in one post by more methods. Each particular interest found in given post is associated with a vector of scores $S = (score_1, score_2, \ldots, score_N)$, where each $score_x$ (user, interest, post, $method_x$) belongs to one of N enrichment methods. Aggregation is a function, which takes vector S as an input, and outputs a score from range <0,1>. We proposed several aggregation functions:

$$Average\ score = \frac{\sum_{x=1}^{N} score_x}{N}, \tag{3}$$

$$Highest\ score = max\ (score_x), \tag{4}$$

$$Accumulated\ score = accScore\ (N), \tag{5}$$

where *accScore(N)* is a function computed recursively as:

$$accScore(x) = accScore(x-1) + [1 - accScore(x-1)] * score_x, \tag{6}$$

where accScore(0) = 0.

AF_p is solving the same issue with only one difference – same interest can be found in more posts. Therefore, the same aggregation functions can be used. If only AF_p is performed, we obtain the user model for each enrichment method. Similarly, if only AF_m is performed, we obtain interest model for each particular post.

3.4 Interest Filtering

In order to increase the quality of user interest model, we use two types of filtration:

- Empirical filtration – filter out false positive interests often repeating in microblog domain (e.g. there is word "RT" mistakenly marked by OpenCalais as semantic web entity).
- Score based filtration – filter all interests with aggregated score lower than given threshold.

4 Evaluation and Conclusion

The main hypotheses of our work are:

1. Aggregation of enrichment methods results in better user interest model, than employing any single enrichment method.
2. Aggregation results in better user model if the interest relevance of posts is considered.
3. Better user interest model is created if low relevant interests are filtered out.

We proposed two methods for evaluation of created user models. The first one is a posteriori *manual annotation*, where human annotators are marking interests as correct or not correct. The goal is to evaluate at least 300 interests for any of enrichment methods, by at least 3 annotators. Based on this evaluation we can compute precision of methods, but also amount of correct interests, not mentioned by user in his posts.

The second method is *synthetic evaluation*, where user posts are divided to train set used to build the model and test set used to test given model, what is a common approach [1, 2]. In our work, posts are ordered by time of creation and divided into 5 equal groups of posts. Four of these groups are used as train set to create a user model, last part is test set – viewed as text representation of its posts. Then, we can compute precision (how many of interests from model have its text representation in test set) and recall (how many of words from test set are found in model). In order to provide significant results, we applied 5-fold cross validation. The final results are averaged from the partial results.

4.1 Dataset

In this work, the UMAP 2011 dataset[2] is used. It is collection of 2 316 204 microblog posts written by 1619 users on Twitter, collected between end of October 2010 and start of January 2011. The results of News enrichment method are already included in the dataset.

4.2 Preliminary results

So far, we have conducted two experiments. Both of them are evaluated only on one random user (i.e., 2000 posts), *synthetically*, with *highest score* aggregation, without *empirical filtration*. Currently, only the Baseline, Youtube, News and Tagdef enrichment methods are implemented. Nevertheless, we can draw some conclusions based on preliminary results we obtained.

In the first experiment, *score* is computed as a product of weight and confidence (λ_1=1, λ_2=1, λ_3=0) and filtered on 50% threshold. Our goal was to compare particular methods with Aggregated method (aggregated Youtube, News and Tagdef method). The results are shown in Table 2.

Table 2. Comparison of enrichment methods.

	Baseline	**Aggregated**	**Youtube**	**News**	**Tagdef**	**Random 1000**
Precision	8.98%	3.57%	3.53%	5.04%	**18.81%**	8.92%
Recall	1.60%	**3.53%**	2.57%	1.46%	0.20%	1.23%
F1	2.71%	**3.55%**	2.97%	2.26%	0.39%	2.16%

The results show that Aggregated method is most successful according to F1 measure. This result supports our hypothesis 1 and point that combination of enrichment method is useful.

Compared to Baseline method, Youtube and News enrichment methods have better recall and worse precision. We believe that the proposed advanced weighting, aggregating and filtering will further increase precision of enrichment methods. Surprisingly good results (comparable to Baseline method) are achieved by Random 1000 method, where 1000 random interests from foreign users were selected as user interest model. This can be caused by relatively short period of collecting data, so the user models are in reality quite similar. This behaviour will be more explained later after performing the second proposed evaluation method.

The comparison of *weight* (λ_1=1, λ_2=0, λ_3=0) and *confidence* (λ_1=0, λ_2=1, λ_3=0) as setups for filtering evaluation is shown in the Figure 1. The x axis is filtering threshold, the y axis is F1 measure for Aggregated method (Youtube, News and Tagdef enrichment methods aggregation).

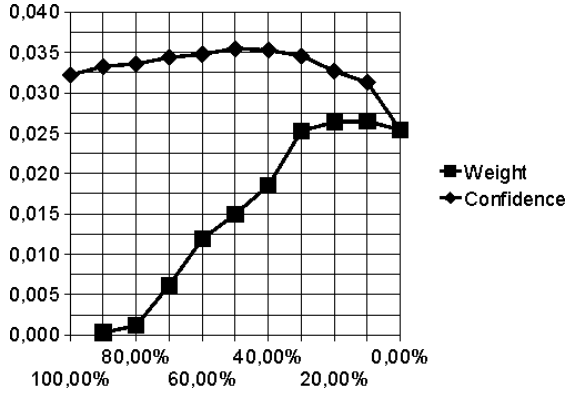


Figure 1. Confidence and Weight comparison. x axis is filtering threshold, y axis is F1 measure.

[2] http://wis.ewi.tudelft.nl/umap2011/

The conclusions of our experiments are:

- if *confidence* is used as the only one weight for score computation, *score based filtration* improves the quality of model (mostly if threshold is set to 50%),
- if *weight* is used as the only one weight for score computation, *score based filtration* can decrease the quality of model. In addition, the quality of such a model is always worse than if *confidence* weight is used.

We proposed method for user interest modelling by incorporating multiple enrichment methods. Preliminary results show that this method has a great potential to improve results of the baseline method. We expect even better results using further advanced weighting and aggregating methods.

Our future work covers evaluation using bigger dataset and more enrichment methods, to find suitable weighting and aggregating schemes. Manual annotation of user interests is necessary to explain good results of Random method in more detail and to further research our hypotheses.

Acknowledgement: This work was partially supported by the Slovak Research and Development Agency under the contract No. APVV-0208-10.

References

[1] Abel, F., Gao, Q., Houben, G., Tao, K.: Semantic enrichment of twitter posts for user profile construction on the social web. In: *Proc. of the 8th extended semantic web conf. on Thesemanic web: research and applications*, (2011), Springer-Verlag, pp. 375–389.

[2] Abel, F., Gao, Q., Houben, G., Tao, K.: Analyzing User Modeling on Twitter for Personalized News Recommendations. In: Proc. of the 19th int. conf. on User modeling, adaption, and personalization, (2011), pp. 1–12. Springer-Verlag.

[3] Bernstein, M.S., Suh, B., Hong, L., et al.: Eddi: interactive topic-based browsing of social status streams. In: *Proc. of the 23nd annual ACM symposium on User interface software and technology*, (2010), pp. 303–312.

[4] Chen, J., Nairn, R., Nelson, L., Bernstein, M., Chi, E.H.: Short and tweet: experiments on recommending content from information streams. In: *Proc. of the 28th Int. Conf. on Human Factors in Computing Systems*, (2010), pp. 1185–1194.

[5] Gao, Q., Abel, F., Houben, G.: GeniUS: generic user modeling library for the social semantic web. In: *Proc. of Int. Conf. on The Semantic Web*, (2011), pp. 160–175. Springer-Verlag.

[6] Kim, Y., Shim, K.: TWITOBI: A Recommendation System for Twitter Using Probabilistic Modeling. In: *Proc. of the 2011 IEEE 11th Int. Conf. on Data Mining*, (2011), pp. 340–349.

[7] Plumbaum, T., Wu, S., Luca, E.W.D., Albayrak, S.: User Modeling for the Social Semantic Web. In: *The 10th Int. Semantic Web Conf.: Semantic Personalized Informaton Management*, (2011), pp. 1–12.

[8] Sakaki, T.: Earthquake Shakes Twitter Users : Real-time Event Detection by Social Sensors. *Proc. of the 19th int. conf. on World wide web*, (2010), pp. 851–860.

[9] Sosnovsky, S., Dicheva, D.: Ontological technologies for user modelling. *Int. Journal of Metadata, Semantics and Ontologies*, 5(1), (2010), pp. 32–71.

[10] Xu, Z., Ru, L., Xiang, L., Yang, Q.: Discovering User Interest on Twitter with a Modified Author-Topic Model. In: *Proc. of the 2011 IEEE/WIC/ACM Int. Conf. on Web Intelligence and Intelligent Agent Technology*, vol. 1, (2011), pp. 422–429.

[11] Yang, Z., Xu, J., Li, X.: Data Selection for User Topic Model in Twitter-Like Service. In: *IEEE 17th Int. Conf. on Parallel and Distributed Systems*, (2011), pp. 847–852.

Personalized Web Documents Organization through Facet Tree

Roman BURGER*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
xburger@is.stuba.sk

Abstract. With vast amount of accessible and relevant information and resources through the Web, one may start to seek for effective archiving and organization of resources. Most existing solutions though support only very specific use case scenarios and are not generalizable to the broad public. We propose a method for personal information management based on facet view of the personal information structure. The structure is displayed in the form of a tree. Facet chaining can create any depth of the structure and thus specify any context of resources. For even easier use, we enhanced this method by semi-automatic clusters extraction of similar resources.

1 Introduction

Based on recent web browsing strategies (for example parallel browsing), new information sources (social and collaborative services) and the sheer fact that the Web is full of possibly interesting information, we can see the problem of information overload. Personal information spaces gets easily flooded with available reading resources. All this resources can still be valid and relevant for particular user. This is especially true in the case of digital libraries. We literary have whole libraries at the reach of our hands, full of quality work.

It then becomes very important to be able to properly archive, maintain and retrieve all this information. Typical frameworks and solutions for organization tasks tend to ignore the fact that user needs may vary greatly between individuals. Even employees of the same domain can have radically different information management strategies as observed in [4]. These strategies are mostly based on personal preferences of individuals, but can also be influenced by events such as vacations.

In this paper, we explore the problem of personal information management. Our goal is to propose new method of organizing and archiving web resources in an effective, easy to use and user friendly manner.

---

* Master study programme in field: Software Engineering
Supervisor: Professor Mária Bieliková, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 125–130.

2 Related Work

Along other researchers such as [11] and existing definitions we identified three basic operations as part of the organization process:

- *archiving new resources* – is a process of expanding personal collection with new resources. Input to this process is the resource itself and metadata describing it, giving it context. It is up to the user how specific the context is (without any additional information or exhaustive specification). Output of the process is resource archivation;

- *retrieving archived resources* – is a process of searching and retrieving sought out resource. Retrieving can be either destructive (resource is removed from the collection) or preserving (resource is kept in the collection). Resource query is the input to the process. Output is usually constructed from the collection of best matching resources to specified query;

- *editing resources* – is a process of updating resource information, usually the metadata and relationships between resources. This process can be actually carried as a series of destructive retrieval and archiving with new information.

There has been extensive research done such as [6] and [1] in identifying main strategies commonly used in personal information management. Most of the works share the idea of a spectrum, where on one side are strategies that rely on almost context free archiving. This strategy tends to be easier to use at first (with reasonable library sizes), but can radically impact effectiveness in large libraries. Resources tends to be harder to find and it is not uncommon to lose resources (completely forgetting about it).

On the other side are strategies that rely on punctual structuring of the personal library. Advantages of fully structured library are in better transparency and also is this strategy less prone to errors and resources losing. Obvious drawback is harder to create functional state of library and more time required maintaining the library. One of the latest researches [3] identified three basic strategies (or roles) that most users can mapped onto:

1. piling strategy,
2. filling strategy,
3. structuring strategy.

Piling strategy is on the context free side of the spectrum and structuring strategy is on the context full side. Filling strategy is somewhere in middle of the spectrum. Filling strategy is though not about using average amount of context to describe resources. Filling strategy is more of a combination. Some parts of the personal library are in context free zone, having stacks or piles of resources that user wants to dig in later (or never). Other parts of the library are reasonably structured, giving the user option to fill in new resources, that are in great importance to the user.

Through the history of personal information management there have been numerous methods, more or less successful. We could organize our resources through printing the resource, sending in the email as an attachment, using permanent browser tabs and windows (in case of the web resource), bookmarking web resources and others.

Two of the most typical recent methods organizing web resources are with bookmarks and tags services. Bookmarks usually utilize folder structure so they are suited for structuring strategy. Problem with maintaining huge structured libraries were tried to solve using information retrieval algorithms such as clustering and classification. Authors in [8] used n-grams in documents to find clusters of similar documents. In [10] authors used incremental clustering to simulate more typical user scenarios. Tags are keywords assigned to a web resource that has special meaning for the

user [5] and the resource can be easily retrieved by the keyword-resource association. Special type of organization is no organization at all, but retrieving by repeated search. This can be achieved with personalized or some advanced search engines like Google using the page rank algorithm [9].

Limitations of these and most other methods are in very low adaptability to organization strategies different then those they were designed for. Bookmarking services are too inconvenient for resources such as "read later" web pages, where users want to archive such a web page as simply as possible. Tags service is in its core structuring strategy (each keyword is closer specifying the context) but does not offer stable transparent library structure required for users using structuring strategy. Also none of the reviewed methods offer easy and simple framework for resources cleaning that is very common for filling strategy. Users usually need to manually edit each unsorted resource and fill it into the right location inside library.

3 Method for Personalized Web Resources Organization

We propose our organization method based on facet filtering. Facet filtering allows us to construct various views on the same subject (our personal resources library). In our domain it means constructing different context views, specifying particular collections of resources.

Facets are non overlapping sets, where each describes particular aspect of a resource. Since they are not overlapping, we can combine them to better specify said resource. The values of the facet are usually extracted directly from collection of resources and the name of the facet describes logical grouping (of non overlapping sets). Each facet describes some portion of the whole context. It is important that each facet should be reasonably defined for its domain and that each resource should be covered by at least one facet [7].

Users normally work with state-full personal organization structures (meaning that structure maintains its internal state until explicitly updated). This is in contrast with typical facets methods, that usually look upon facets as querying framework. Therefore in our design, we utilize new concept of facet tree originally proposed in [12]. Facet tree maintains its state and can be easily dynamically adjusted. Individual facets in chained facet tree can be removed or added creating context views on demand (and can still perform as a search tool). To our best knowledge, this concept has not yet been used for web documents. Nonetheless web resources greatly benefit form this design because of the broad nature of web resources. Short term and long term resources often group together so dynamic library structure is desired. The prototype of facet tree interface is shown in Figure 1. Example shows chained facets *Color* and *Author* and the respective dynamically generated hierarchical tree. Any of the chained facets can be removed anytime and new facet can be added to the tail of facet chain. If any of the documents do not have required metadata for particular facet, facet is ignored for the document (Figure 1 shows document authored by X but with no associated color).

Facet tree makes personal information management easily available for broad range of user roles and strategies. It does not though helps much in terms of effectiveness. For example, if we have two documents that we want to keep together, we have to find shared facets (most likely archivation time). But if we can't, we have to invest more time and effort into organization. This usually happens if we want to achieve some higher level of abstraction then available facets offer. Such example would be *Project* name folder.

We enhance facet tree with semi-automatic cluster extraction of similar resources from selected collection. If user finds a large collection of resources that can not be properly structured with facets, user has option to let the method extract clusters of resources. The data for clustering algorithms come from all metadata available from resources in selected collection thus possibly creating very accurate clusters of resources. To ensure validity of clusters over time and addition of new resources, we will combine cluster extraction with classification of new resources. Classification will determine if the new resource is valid member of any previously extracted cluster.

Generated clusters need to be properly and consistently stored in personal library. Facet tree though has dynamic nature. Therefore we are defining new, special facet that will host generated

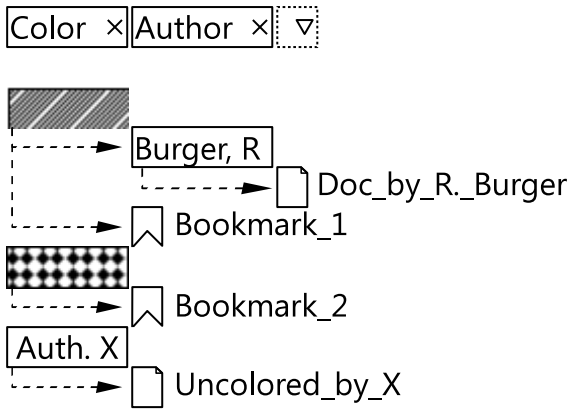


Figure 1. Facet tree with chained facets Color and Author.

clusters. We call this new facet *Color* and values are going to be defined colors. As stated earlier, the concept of a *Color* is supposed to represent a higher level of abstraction. We chose *Color* because it does not require any addition user input and users can still have their own association in mind between *Color* facet and collection of documents represented by the color. Each new generated cluster gets assigned one unused color. Our method works in following steps:

1. User selects a collection of documents that needs to be cleaned up.
2. We analyze and create clusters based on available metadata. Each cluster gets assigned one of the unused color.
3. User can manually adjust documents in clusters if required.
4. Newly archived documents will be checked whether they can be filled into existing clusters using classification algorithm (along with extracting metadata for facet tree).
5. When collection of documents belonging to any cluster (or any other facet) gets to large, the process can be repeated from step 1.

We have designed our solution to meet most of the common user needs and use cases in their information management strategies. Table 1 shows how each most common user actions are mapped to actions of our proposed method and reflects how the result can be achieved using our proposed method. Actions of different strategies are in Table 1 logically separated.

4 Evaluation

For evaluation of proposed method are focused on web resources, specifically research papers in digital libraries. We have integrated our organization method to the web bookmarking and annotation service Annota [2] as one of the views on the personal library. Annota lets users archive their resources by tags or in folder structures and create annotations or highlights in documents. Annota also has collaborative dimension, allowing for creating groups and sharing personal bookmarks with colleagues.

Currently there is 98 registered users with together 4544 archived documents. Users assigned 1339 tags and so far created 48 groups. In a month of functional folders archiving feature (without any special promotion), 29 users implicitly created their root folder (by visiting the subpage with this feature) and 7 explicitly created at least one folder. Maximum number of folders for one user is currently 5.

Table 1. Information management strategies actions mapping.

User action	Proposed method
Piling strategy	
Simple resource archivation.	Only one click is needed to archive resource.
Resources listing based on archivation time.	Select *Added* facet.
Resources search.	Select facet and input search criteria for its values.
Filling strategy	
Archive resource by filling into structure.	Adding new resource automatically extract metadata. If some more abstract context is required, user can use *Color* facet.
Personal library browsing.	Chain facets as required.
Cleaning up large collection.	User selects resources and automatic clusters extraction is performed.
Structuring strategy	
Upfront structure creating.	No action needed.
Archive resource.	Adding new resource automatically extract metadata.
Navigation to specific resource.	Chain facets as required.
Resources search.	Select facet and input search criteria for its values.

5 Conclusion

In this paper we proposed using new concept of facet tree for personal information management in the domain of web resources from digital libraries. Our contribution to this field of research is in enhancing this concept with semi-automatic clusters extraction from specified collection of resources. We seamlessly integrated this new feature with facet tree by special facet category – the color. Our aim was to achieve simple and clean framework for information management, easy and effective to use for most users. We took great care understanding different information management strategies and made sure we support most common organization scenarios.

We plan to fully evaluate proposed method in quantitative as well as qualitative experiments. In quantitative experiment we compare our method with traditional bookmarking service based on structuring personal library (also integrated to Annota). We compare various statistics logged in user sessions, trying to evaluate if Annota users can really be mapped onto identified strategies and whether proposed method is indeed useful.

We have designed qualitative experiment to take place in two sessions. In the first session users are asked to find and archive selected pieces of information. Second sessions is planned at least two weeks after the first. In the second session, users are asked to find some bits and parts of information from previous session. We will closely examine if so and how they proceed with these tasks using proposed method. We will identify weak and strong points of our design and deduce conclusions (and possibly feature improvements).

Annota is also helpful with validating our enchantment to facet tree, the cluster extractor. Since Annota offers folders structuring for resources there is source of data on how users setup relations between theirs resources. In quantitative experiment we then compare it to clusters created in simulation of particular users where we input the clustering method various sub-collections from said user library.

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/0675/11.

References

[1] Abrams, D., Baecker, R., Chignell, M.: Information archiving with bookmarks: Personal Web Space Construction and Organization. In: *Proc. of the SIGCHI conf. on Human factors in computing systems - CHI '98*, New York, New York, USA, ACM Press, 1998, pp. 41–48.

[2] Ševcech, J., Bieliková, M., Burger, R., Barla, M.: Researcher activity tracking enriched with annotations in research papers from digital libraries (in Slovak). In: *Proc. of the 7th Workshop on Intelligent and Knowledge Oriented Technologies - WIKT '12*, Smolenice, Slovakia, Nakladateľstvo STU v Bratislave, 2012, pp. 197–200.

[3] Henderson, S.: Personal document management strategies. *Proc. of the 10th International Conference NZ Chapter of the ACM's Special Interest Group on Human-Computer Interaction - CHINZ '09*, 2009, pp. 69–76.

[4] Kelly, D.: Evaluating personal information management behaviors and tools. *Communications of the ACM*, 2006, vol. 49, no. 1, p. 84.

[5] Lee, K.J.: What goes around comes around. In: *Proc. of the 2006 20th anniversary conf. on Computer supported cooperative work - CSCW '06*, New York, New York, USA, ACM Press, 2006, p. 191.

[6] Malone, T.W.: How do people organize their desks?: Implications for the design of office information systems. *ACM Transactions on Information Systems*, 1983, vol. 1, no. 1, pp. 99–112.

[7] Mas, S., Marleau, Y.: Proposition of a Faceted Classification Model to Support Corporate Information Organization and Digital Records Management. In: *2009 42nd Hawaii International Conference on System Sciences*, IEEE, 2009, pp. 1–10.

[8] Miao, Y., Kešelj, V., Milios, E.: Document clustering using character N-grams. In: *Proc. of the 14th ACM int. conf. on Information and knowledge management - CIKM '05*. Number 2, New York, New York, USA, ACM Press, 2005, p. 357.

[9] Page, L., Brin, S., Motwani, R., Winograd, T.: The PageRank citation ranking: bringing order to the web., 1999, pp. 1–17.

[10] Sahoo, N., Callan, J., Krishnan, R., Duncan, G., Padman, R.: Incremental hierarchical clustering of text documents. In: *Proc. of the 15th ACM int. conf. on Information and knowledge management - CIKM '06*, New York, New York, USA, ACM Press, 2006, p. 357.

[11] Teevan, J., Jones, W., Capra, R.: Personal information management (PIM) 2008. *ACM SIGIR Forum*, 2008, vol. 42, no. 2, p. 96.

[12] Weiland, M., Dachselt, R.: Facet folders: flexible filter hierarchies with faceted metadata. In: *Proc. of the twenty-sixth annual CHI conf. extended abstracts on Human factors in computing systems - CHI '08*, New York, New York, USA, ACM Press, 2008, p. 3735.

Preprocessing Linked Data in Order to Answer Natural Language Queries

Peter MACKO*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
pmacko@outlook.com

Abstract. Searching for information on the Web is difficult because of its enormous growth and web content is in unstructured format. To utilize the full power of the structured data a special query language, like SPARQL, has to be used. However, it requires knowledge of special syntax and it is difficult for common users. In this paper we propose a method for transformation of queries in pseudo-natural language into SPARQL. The most important part of our method is pre-processing. Our method creates two lexicons that help us translate user queries to dataset dictionary. We bring weighting system that helps us determinate what user mean.

A paper based in part on this paper was submitted to World Wide Web, Internet and Web Information Systems (WWW 2013), Springer.

* Master degree study programmer in field: Software Engineering
Supervisor: Michal Holub, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

Query Building Method for Texts Similarity Detection in the Web Resources

Pavel MICHALKO*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
michalko.pavel@gmail.com

Abstract. In this paper we introduce several methods for search query building. This is an important step in whole process of texts similarity detection among the sources on the Web. Query is considered as a fingerprint for given part of text. It can be afterward used as an input for Web search engine to obtain relevant Web sources. All of these methods are primary aimed on use with Slovak texts and they are based on specifics of this language. The main purpose of this paper is to discover the most effective approach for selection words that will represent search query.

1 Introduction

Plagiarism can be defined as intentional or unintentional copying of other authors work without correct citation of it. Actually, if only some parts or a part of original work was copied in that way, it is still qualified as a plagiarism. It is also important to explain expression "copying" in relation to plagiarism. Generally, when authors are aware of work that is plagiarized, they are trying to modify it. The reason is to hide similarity with original work. Therefore copying can be also considered as misusing of original author's idea or ideas.

This problem becomes more expanded in nowadays era full of information technologies [6]. Main role here belongs to the Internet. It is very helpful in information retrieval and everybody are able to find needed sources effortless. However, this has negative impact on creation of new works as well. Many people will rather use this easier way instead of spending more their time to write document or paper in the right way. And there is the heart of the whole problem.

The idea of new work creation is to acquire new knowledge and in best case to enrich it with own addition. As we can suspect, the most affected are schools and field of research. Although there are other problematical domains e.g. mass media or Web content.

In last days the problem of plagiarism became more discussed. The cause was hype around people with higher social status that copied part of their thesis. Prevention of plagiarism is strongly recommended and can avoid that kind of problems in future.

Searching for similar documents on the Web manually is very time consuming. Even more if we need to analyse larger or more documents at once. Therefore it is necessary to automate whole

* Master degree study programme in field: Software Engineering
Supervisor: Assoc. Professor Daniela Chudá, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 137–142.

process of text similarity checking in the Web resources. The main problem is to decide in which way similar documents from the Web will be searched. For this purpose a search query for text is need to be made. It is important to make this properly, because based on this query only we obtain results from Web search engine. We based our work on Slovak language. Decision we made was result of our research, in which we discover that there is no fully working tool for checking Slovak texts similarity among the Web sources yet.

2 Related work

In [2] was introduced an approach that is based on 7-Grams. First, all of 7-grams from text are made. After that, 5% of them are randomly chosen. This is an effective way to made search query. On the other hand, the number of needed search queries is too high. That can lead to faster search engine quota using.

Authors in [1] were focusing on search query building methods. This work is very useful for our purposes and it serves as a basis for this paper. Several approaches to query building were introduced there such as selection of facts, nouns, and the most frequented words.

Similarity detection on the Web is related to query building problem. In [4] most of the popular tools were introduced and described. All of these tools are commercial and does not have detailed documentation. As a result of that we failed in obtaining information involving approach to query building. From plenty of these tools we can mention e.g. *Turnitin [3], SafeAssign, CheckForPlagiarism.net* and others.

Free available tools such as *DOC Cop* are often based on not so sophisticated query building method. The N-Gram selection technique is performed for given document. Final number of created queries for this document approximately equals the number of words in entire document. This deficiency is similar to previously described problem appearing in [2].

The main problem of all these tools is absence of correct Slovak language support. Difficulties appeared in texts with special characters like letters with diacritics, when the similarity detection may fail. Even if process of similarity detection ended, problems with Slovak texts encoding result to incorrect similarity level identification.

3 Web search

For texts similarity detection we can choose from two types of information sources: local corpus of documents and the Web. There are a lot of differences between these two types.

Searching in local corpus is faster, more reliable and easier than on the Web. Reliable means that working with Web sources can be affected by more factors than with local corpus (e.g. server's accessibility or network state). We can simplified say that similarity detection on the Web is like in local corpus with additional steps. It means that more effort is needed in case of similarity search among the sources on the Web. As we mentioned earlier, using the Internet to find needed information's sources is very simple and available. For that reason still more and more people are using the Internet as source of information.

3.1 Web search engines

To find similar documents on the Web we first need a suitable mechanism to obtain relevant Web sources. There are two ways how to achieve that: build own Web index or use one of the existing tools. Nowadays building own index does not make too much sense. There was a lot of effort expended on Web search engine development. Existing tools are more reliable and tweaked. The most known are Google, Bing and Yahoo!. To use these commercial engines it is needed to pay. On the other hand it is still way how to use them for free with some restrictions [1]. Other option is to use free Web search engine like DuckDuckGo.

It is very important to know, how given search engines are behaving. We have to take in mind all of these characteristics in query building process. Based on our Slovak Google Web search engine experiment we can state for search query that:

1. It depends on word's order (two queries consisting of same words but in different order will return different results).
2. Stop words (i.e. prepositions or conjunctions) are omitted.
3. Maximal length of search query should be not more than 10 words [1].
4. Synonyms are considered as different words.

3.1.1 Comparison

The question about differences between free and commercial tools is here on place. We performed an experiment to determine this. Five different Slovak texts were copied from five Web sources. After that we made search queries and got results by mentioned Web search engines. For query building we used some existing methods from [1] (where FNF means combination of facts, nouns and most frequented words and random means selection of random words).

Google achieves best results and nowadays the most of people are using it. That's why it is the best solution for searching similar Slovak texts on the Web right now.

DuckDuckGo, as a free tool representation, does not have so good results. On the other hand it is free and we can make incomparable more searches than with Google. The decision on which Web search engine will be used depends on specific application's requirements. Naturally the combination of several search engines is also possible in depending on their actual quotas.

4 Query building

Before we can take advantages of some Web search engine, it is logically necessary to find the best way how to create an input for it. For this purpose search queries have to be created.

What search query actually stands for? In fact it is text string consisting of key words for given text. The better method of creation search query will be, the better results will be obtained. Just as we mentioned in the introduction, quality of produced query is crucial. It will be used as fingerprint for analysed text and only one information about text for Web search tool.

4.1 Text chunking

On the beginning, it is mandatory to point out that we made decision to work with parts of text separately rather than with the entire text at once. There are few methods to divide text to smaller parts – chunks [5]. However, this work is based on assumption that one text's paragraph should express one idea. If someone wants to use the Web as a resource in his or her work it is more probable that this source will be used right for paragraph. Therefore we are suggesting to split text by paragraphs. If too long paragraph in text will appear, it may be divided to smaller chunks. The best way is to choose words count threshold. This text chunking based approach will reduce the number of needed search queries for given document, which value is equal to count of text's paragraphs.

4.2 Our methods

In [1] were introduced certain query building methods. Combination of facts, nouns and the most frequent words (FNF) was selected as the best one. We created several different query building methods based on Slovak language specifics. These methods are:

- *Subordinate clause:* words from second parts of sentences, which starts with comma.
- *Sentence's borders*: first and last words of sentences.

- *Middle of sentence*: words from the middle of sentences.
- *Longest words*: words with the largest number of characters.
- *Facts (own approach)*: numbers or words starting with capital letter (excluding beginning of the sentences).
- *Random sentence*: sequence of words from random selected sentence.
- *Longest sentence*: combination of longest words and facts selection from the longest sentence in paragraph.

In case of subordinate clause, sentence's borders and middle of sentences methods we wanted to figure out if position of words have some weight in term of search queries. Selection of the longest words was chosen considering length of word as potential important. Random sentence and facts approach were based on random word selection and facts method used in [1]. Last mentioned longest sentence method was created due to assumption that this sentence is containing paragraph's main idea.

The number of words in used search queries is not strict given and variable for every chunk. It depends on used query building method or length of given text chunk. The limit for longest word selection method was set to maximum 6 words and for rest of methods was this limit set to 10 words. The final number of words in created query can be lower if number of words that were matching condition of selection for given method is less than this pre-set limit (e.g. chunk consisting of less sentences). The selection of words is now performed in order how they appear in text. Although taking in mind the order of words in process of query creation maybe can improve results in future.

4.3 Experiments

To confirm our hypothesis for introduced query building methods we performed an experiment. It is important to note that all experiments we made were performed on Slovak texts. Analysed text was literally copied random chosen article from portal `www.sme.sk`. Afterward we created queries by every method that we suggested. The comparison was elaborated against the best method (FNF) and evaluated by tool from work [1]. We used Google as Web search engine in all experiments for its best achieved results. Results of this experiment are shown in Figure 1, where achieved similarity level for every method is presented.

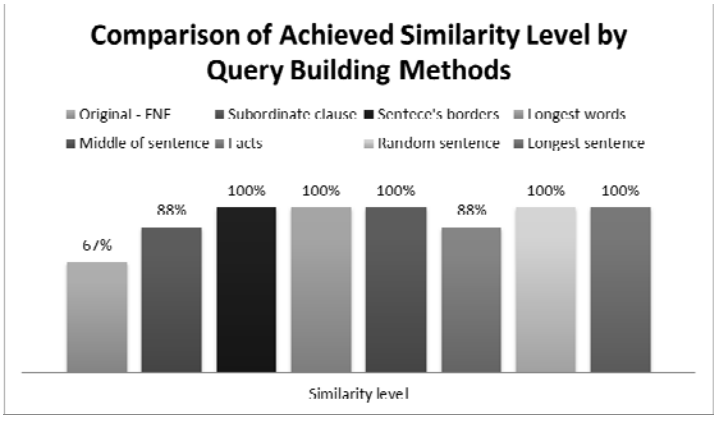


Figure 1. Comparison of Achieved Similarity Level by Query Building Methods.

After that we performed another experiment to reduce inaccuracy. This time we used longer and slightly modified student's paper to simulate plagiarism. Original paper was supplemented with

part of other student's paper. The order of some words was changed and several words were also replaced with synonyms as well. Document has been divided to 25 chunks and for every chunk we created query using all methods. This time we omitted own approach for facts extracting for the reason that similar method was already verified in [1].

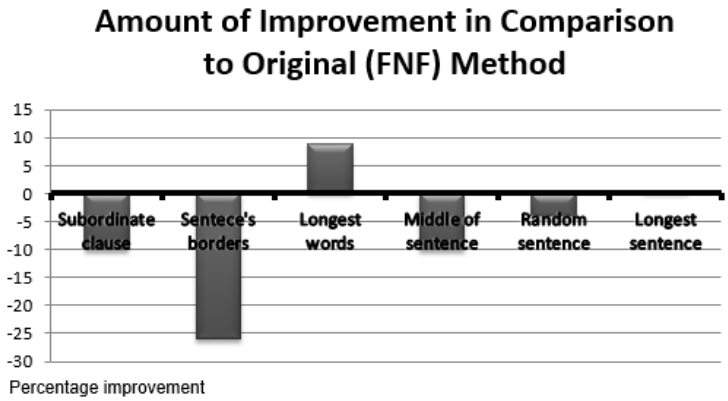


Figure 2. Amount of Improvement in Comparison to Original (FNF) Method.

4.4 Experiments evaluation

Depending on performed experiment's results we can state that the best search query building method in term of Slovak texts similarity detection is selection of longest words from paragraphs. Achieved results were better than in case of original method. Our assumption is that frequency of longest words is not as high as in case of shorter words and they are also more unique. This indicates that rarely occurred words with higher length have bigger importance in text chunk representation. We also discover that optimal length for this kind of query is in range from 4 to 6 words. When we used more words, the bias in results started to appear.

Another suitable method seems to be using the longest sentence in paragraph. Results are equals to original compared method. This approach can be used as basis for different query building methods. It can be improved with few optimizations or more appropriate word selection.

Results of second experiment show that we are able to find similar Web sources even if analysed text was modified from original and not copied literally. It is also obvious that the amount of copied text can affect final result. If only few words in large paragraph will be copied, it is possible that these words will be not selected to search query. That can lead to original sources omitting.

5 Feature work and conclusion

Naturally, there is always place for improvement. In our case we currently see this possibility in:

- Creation of new methods for query building.
- Finding better combinations of existing methods.
- Optimization of Web searching and sources retrieval.
- Building query that is optimized for modified (not literally copied) texts.
- Method's experiments comparison with English texts.

We have described several issues around texts similarity detection in the Web resources. That involves problems such as query building, text chunking and obtaining relevant Web sources. Our main goal was to develop better method for search query building compared to the existing best

one. Experiment's results shown that we accomplished this with building query from the longest words.

We also have mentioned and explained the problem of plagiarism and importance of its solution. Each of our suggested solutions was based on actual research as much as possible to utilize existing knowledge.

Suggested methods are very effective in regards to quantity of needed requests to Web search engine. Standard query building techniques are based on creation plenty of queries from given document. That is more time consuming and restrictive due to Web search engines query quotas. Our approaches based on chunking are reducing this number of created queries for the entire document with preserving capability for obtaining relevant Web sources. Using these methods is suitable for plagiarism detection on the Web.

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/0971/11.

References

[1] Bieliková, M. (ed.), Lačný, J., Maršalek, M., Michalko, P., Súkeník, J.: Plagiarism Detection on the Web. *Student Research Conference*, Bratislava, Slovakia, (2012), pp. 481–482.

[2] Butakov, S., Shcherbinin, V.: On the Number of Search Queries Required for Internet Plagiarism Detection. *Ninth IEEE International Conference on Advanced Learning Technologies*, (2009), pp. 482–483.

[3] Jones, K. O., Moore, T. A.: Turnitin is not the Primary Weapon in the Campaign Against Plagiarism. *Proceedings of the 11th International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing on International Conference on Computer Systems and Technologies*, New York, USA: ACM Press, (2010), pp. 425.

[4] Kakkonen, T., Mozgovoy, M.: An Evaluation of Web Plagiarism Detection Systems for Student Essays. *Proceedings of the 16th International Conference on Computers in Education*, (2008), pp. 99–103.

[5] Pataki, M.: Plagiarism Detection and Document Chunking Methods. *The Twelfth International Word Wide Web Conference*, Budapest, Hungary, (2003).

[6] Pločica, O., Telepovská, H.: Metódy detekcie plagiátorstva. In: *Proceedings of UNINFOS2009*. Nitra, Slovakia, (2009), (in Slovak).

Related Documents Search Using User Created Annotations

Jakub ŠEVCECH*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
sevo_jakub@yahoo.fr

Abstract. We often use various services for creating bookmarks, tags, highlights and other types of annotations while surfing the Internet or when reading electronic documents. These annotations can be used to support navigation, text summarization etc. We proposed a method for searching related documents to currently studied document. Proposed method uses annotations created by the document reader as indicators of user's interest in particular parts of the document. The method is based on spreading activation in text transformed into graph. For evaluation we created a service called Annota, which allows users to insert various types of annotations into web pages and PDF documents displayed in the web browser. We analyzed properties of various types of annotations inserted by users of Annota into documents. Based on these we evaluated our method by simulation and we compared it against commonly used TF-IDF based method.

Amended version published in Information Sciences and Technologies Bulletin of the ACM Slovakia, Vol. 5, No. 2 (2013), pp. 44-47.

* Master degree study programme in field: Software Engineering
Supervisor: Professor Mária Bieliková, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 143–148.

Exploratory Search on Twitter Utilizing User Feedback and Multi-Perspective Microblog Analysis

Michal ŽILINČÍK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
michal@zilincik.eu

Abstract. In this paper we propose an approach to exploratory search on Twitter which deals with the issues connected to this domain. First of all, it is necessary to address information scarcity and low language quality. To offer more complex view of the content, we propose the usage of various metadata as important measures when characterizing microblogs. We are trying not to assume too much about user's intents prematurely, which means not only to analyse subtopics but also common tweet and author metadata, affect in tweets and to take into account the nature of referenced content, if any. The user interest is observed using explicit and implicit feedback.

1 Introduction

Microblogs represent popular way of sharing information on social web. The content has heterogeneous nature and is often created without regard to language or information quality, which poses a significant issue for deeper analysis related to information retrieval tasks.

Tweets offer 140 characters for text and with inclusion of links and mentions, this space rapidly shrinks and so does the efficiency of text processing. Moreover, use of slang and spelling errors are generally observed in informal communication which also takes place on Twitter. To enrich textual content and to better describe documents, retweet count, author's follower count or other metadata provided by Twitter are often used in microblog analysis but do not have to actually be enough to describe user's interests or preferences.

In general, exploratory search means exploring for information in order to learn about a topic. It usually involves an iterative process of exploration and as user learns more about the topic (or related subtopics), their goal is refined as opposed to direct search for facts where information goal is well-defined [5]. Exploratory search can be carried out in different manners in terms of user interface and interaction model. There was a publicly available experiment TweetMotif [8] where the basic idea of exploratory search is topic summarization, which serves as our inspiration. Text summarization usually describes a process in which a document or set of

* Master degree study programme in field: Information Systems
Supervisor: Professor Pavol Návrat, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 149–154.

documents is transformed to significantly shorter version in order to briefly characterize them. However, we understand the term *topic summarization* as the process of extracting the most common phrases from set of documents. It can be viewed as summarization of text down to phrases but with the difference of categorizing the documents into topics based on occurrences of extracted phrases.

2 Related work

To the best of our knowledge, there is no exploratory search tool for Twitter that tries to learn user preferences and estimate their intent while allowing them to explore content of tweets. However, there has been extensive research of partial tasks that the proposed exploratory system incorporates.

When it comes to summarization, work in this area is diverse. To give a basic overview of directions of research, in [4] authors summarize tweets by assigning them a topic label such as politics, technology, sports or entertainment. This method is suitable for exploratory search since user can start exploring the content without even specifying the initial query. Extensive comparison of existing and proposed algorithms for trending topics summarization has been done in [3]. This is the case of common text summarization; tweets for trending topics are known and the task is to show user a tweet that sums up given topic the best. The proposed methods show promising results that can potentially make exploratory search more time-efficient for user. A different approach is shown in TweetMotif [8]; it categorizes tweets into subtopics which are extracted with respect to their occurrence. Chosen phrases characterize corresponding tweets because phrases that do not often occur on Twitter are assigned higher importance.

To make summarization or similar methods more successful, numerous techniques are known that deal with low language quality and information scarcity in tweets. Authors of [4] study lexical-based enrichment to increase features using character n-grams, word n-grams and orthogonal sparse word bigrams. They also describe techniques for overcoming feature scarcity. Link-based external enrichment consists of using words from expanded forms of URLs that are included in tweets. In addition, by part-of-speech tagging and identifying nouns they hope to get better understating of the message topic. Phrases can sometimes be referred to by not using every word from them, so authors explore methods that would for example expand “celtics” found in tweet to “boston celtics” which provide much more information about the topic. An extensive work in area of lexical normalization for short text message is elaborated in [2]. Authors consider many different types of out-of-vocabulary (OOV) words and describe method combining existing and proposed techniques in order to effectively deal with most of the OOV words. Missing or extra letters in words are the most common problem in tweets, followed by so-called internet slang (abbreviations like “lol”) and word or part of the word substitution with numbers that sound alike.

There have been a lot of attempts to evaluate tweet quality and interestingness based on metadata associated with tweets or their authors. In addition to the most straightforward indicator retweet count, influence of URL, mention and hashtag counts along with tweet author follower count or listed count on tweet popularity have been researched [11]. New characteristics based on these measures were proposed as well, e.g. FollowerRank [6]. Measures describing affect have also been used, namely ANEW [1] and related work AFINN [7] developed especially for Twitter.

To make better recommendations, authors of [9] also measure similarity of tweets from user’s timeline to currently analyzed tweets. On social network, authors’ influence on topics and authors’ topical interests can be measured easier than on the web. TwitterRank algorithm [10] based on PageRank quantifies interestingness of content of authors that user follows by utilizing social network structure and topical influence of authors. In another words, it estimates whom user follows because of interesting content and whom because of different (e.g. social) reasons.

The most of the work has been done in area of metadata (meaning non-textual measures) and improving information quality. On the other hand, we feel that there is a lack of research regarding

exploratory search specialized for Twitter. Moreover, most of the mentioned methods try to find mainly static rules for tweet interestingness estimation. We believe that exploratory search needs to respect user's interests as it changes throughout the exploration and focus on user's current intent. We operate with premise that interests differ from user to user and from topic to topic, as well. Predefined rules of what is interesting cannot satisfy everyone every time they look for content. At first, research usually focus on quality of results so many successful techniques described here are not usable in real-time conditions or, as comparing tweets recommended to user by Twitter to new content, they proved useful but still serve more as recommenders than support for exploration of topics not usually read by user.

3 Proposed concept

Following the conclusion from section 2, we decided to focus on exploratory search that allows user to get a sense of what kind of content is being published on Twitter in regard to given topic and customize shown results with respect to what catches user's interest at the time of exploration.

To build such system in reasonable scale, it is necessary to analyze content in real-time. With performance in mind, many decisions need to be made so that acceptable responsiveness is achieved. We focus on practical utilization of existing methods, their modification and integration into system that is able to accomplish the defined goals of exploratory search.

3.1 User interface and interaction model

Faceted browsing is familiar but with scarce content, facets could only allow user to filter by criteria such as URL occurrence, which we believe would not be very useful. That is why we suggest to let user read tweets and explore content referenced by links while the system is quietly gathering implicit feedback and offers a possibility to provide explicit feedback. If relationships between the chosen measures and feedback are discovered, suggested content is offered to user.

The proposed system is based on topic summarization as described in section 3.2. After user enters the topic, they are presented with extracted subtopics and can explore tweets that discuss the given subtopic. To better support exploration, we added two additional actions for user who can choose to load more tweets that include the given phrase or they can decide to explore the subtopic, in which case even more tweets containing the subtopic are loaded in the background, divided into subtopics and then the current subtopic list is replaced by these sub-subtopics. Such drill-down allows for more thorough exploration.

User can provide explicit feedback by rating tweets or whole subtopics using a scale with five stars. On the other hand, implicit feedback is collected based on following observations. Attention time for a tweet is calculated as total hover time on rectangle containing the tweet. User is encouraged to hover the tweet as it becomes more readable due to border, background and text colors. Attention time for a subtopic is currently calculated as sum of attention times of tweets that belong to the subtopic and attention time for referenced content is defined as the time spent between the mouse click on the link and the next user interaction on our system's webpage. Click-through itself is explicitly logged. Count of load-more actions is recorded as well as the act of exploring subtopic when user drills down the subtopic by clicking the explore button.

Behavior of proposed system is shown in Figure 1 and current implementation description follows. Both implicit and explicit feedback is collected while user is exploring content. When user rates at least 10 tweets, a model is trained using SVM (support vector machine). Input consists of attention times for tweet and referenced content and fact whether user clicked on the URL. Instances are classified as either interesting (rated by 4 or 5 stars) or not interesting (1 or 2 stars). Error is measured using K-fold cross validation where K equals the number of explicitly rated tweets. If more than 75% of instances are classified correctly, we move on to the next step.

For tweets that user read but did not rate, classes are estimated (interesting or not interesting). This way, explicit feedback, where missing, is predicted based on implicit feedback.

Afterwards, second model is also trained using SVM where input consists of metadata and subtopic assignments of tweets and classes are the same as in previous case. Now using the second model, interestingness of unseen tweets is estimated and shown to user. We plan to implement mechanism for continuous control of model success rate and functionality ensuring that most of the content shown to user consists of content estimated to be interesting based on the SVM model.

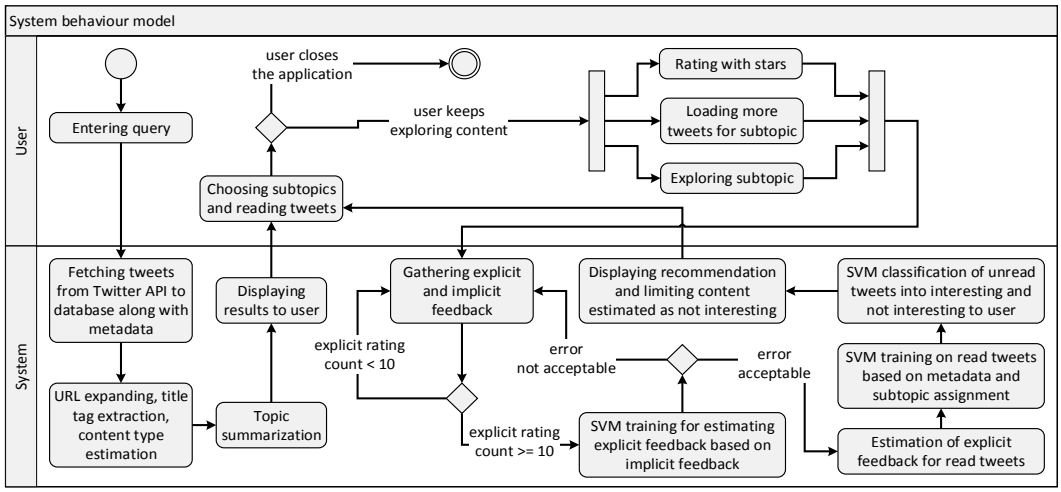


Figure 1. Activity diagram describing basic usage of proposed system.

3.2 Topic summarization

The proposed system is based on topic summarization as implemented in TweetMotif [8] because we consider it probably the most developed exploratory search system for Twitter that also has been deployed as public web application. It showed that summarization is helpful for exploratory search, either as a starting point or as support during the exploration process.

To identify which n-grams represent topics, the extracted phrases are scored by likelihood ratio from [8] which is also used for sorting topics when shown to user. The general tweet corpus used for penalization of commonly used phrases contains almost 200,000 tweets that we collected in February 2013 using search queries for words "the" and "of". The next step consists of merging similar topics as described in [8] but new labels for merged topics are currently picked randomly.

3.3 Fighting information scarcity and low language quality

As a result of very limited space for textual content, literally every word in a tweet counts. As suggested in [4], tweets often include URL. After expanding the shortened URL, more words relevant to the content of web resource can be extracted. But sometimes URLs do not include any meaningful terms, so we decided to overcome this shortcoming by looking into the page content. Consider this URL `http://mahotellaqueens.com/deals/230922493184.html` which was found on Twitter and the title of referenced web page "*FOLK COSTUME BLOUSE Europe Slovakia Ethnic Gypsy Retro Hand Embroidered Kroj $140.00*". Text extracted from HTML title tag clearly tells more than URL itself, so it is appended to tweet text when performing topic summarization and affect analysis. Analysis of the whole content of web page could bring more information but takes significantly more time than just grabbing content of the title tag.

The study [2] dealing with OOV words in Twitter messages proposes to first look up words in slang dictionary, which accounts for biggest increase in success rate, and if no match is found, to use commonly used spellchecking. Derivation of this method is considered for proposed system. Slang dictionary lookup is planned using the Internet Slang Dictionary & Translator available at

`http://www.noslang.com/dictionary/full/`. We also consider using PHP pspell functions to perform traditional spellchecking in the second phase of the described process.

3.4 Measures for describing tweet content

Generally, we believe basic metadata affiliated with tweets can support exploratory search. Sometimes, user can be drawn to tweets with URLs because they may want to find resources concerning their area of interest. On the other side, when looking for overview of people's opinion of some fact, occurrence of URL may not be important as much as retweet or author's listed count.

For describing tweets we observe retweet, URL, mention and hashtag counts. For authors of tweets we observe tweet count, listed count and FollowerRank [6]. We plan to use age of user profiles in conjunction with different measures. Listed or tweet count may tell more if they are analyzed with consideration of how long the user is present on Twitter. We are unable to use many of promising measures. Popularity score [11] cannot be calculated in real time or efficiently obtained in advance and TwitterRank [10] focuses on recommendation based on user's previous activity on Twitter which may not help at all when dealing with topics unknown to user.

When observing affective aspect of tweets, we implemented calculations of valence, arousal and dominance based on ANEW [1]. Many tweets however show no affect according to this list of words so for estimating valence the microblog-optimized AFINN [7] is used as well. In addition to these two word lists, we developed regex expressions for extracting almost every commonly used emoticons. Grouping emoticons expressing the same or similar emotions allowed us to estimate valence by mapping noun or adverbs describing matching emotions to their valence values in ANEW and AFINN. Preliminary results show that many tweets contain emoticons while no word is matched against these word lists, thus making affective aspect acquisition more successful.

We try to categorize URLs based on the nature of referenced content. Many tweets reference photography, audio and video sharing services. We have implemented few simple rules that match domain and other parts of URL, classifying links into image (Twitter and Facebook links to photographs, Instagram, Pinterest, Tumblr, Flickr), video (YouTube, Vimeo), audio (SoundCloud) or unknown category. We think it would be useful to detect links to articles but that would only be possible using comprehensive page catalog or quick intelligent content analyzer.

4 Evaluation

Evaluation of recommendation methods is a difficult task. It does not involve validation against one truth because the correct results are subject to the interests of different users. For that reason, we have to obtain user's opinion separately from what we observed using the proposed method. To measure precision and recall (and F-measure), we will ask user to provide explicit feedback on few tweets. Those tweets will be picked from two groups: read but not rated tweets to evaluate model used for estimating explicit feedback based on implicit feedback and unread tweets to evaluate model for estimation of user's preferences based on metadata and relevancy to extracted subtopics.

To decrease dependence of evaluation on user's willingness to provide such feedback, we will test a different approach. Since all content estimated to be interesting is presented to user and we already observe their feedback, we can estimate precision. On the other hand, along with content estimated to be interesting we will include few tweets that we believe user will find uninteresting but present them as recommended. Based on feedback on those tweets, to some degree we can estimate recall without unnecessary demands on user.

5 Conclusion and future work

In this paper we presented an approach to exploratory search on Twitter as a work in progress. Our method addresses issues connected to nature of microblogs by gaining the most out of textual

content, referenced content and metadata connected to tweets and their authors. We are building system that enables faceted-like exploration of subtopics while not forcing user to provide explicit feedback too often. We focus on silent observation of user behavior and use measures as means for classification of tweets into interesting and not interesting to user in the current session.

In the near future, the user preference model based on user feedback will be the main concern of research and development and will be subjected to empirical testing to optimize the system before final evaluation of proposed approach.

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/0971/11 and the Slovak Research and Development Agency under the contract No. APVV-0208-10.

References

[1] Bradley, M., Lang, P.: *Affective norms for English words (anew): Stimuli, instruction manual and affective ratings.* Technical report c-1, University of Florida, (1999).

[2] Han, B., Baldwin, T.: Lexical Normalisation of Short Text Messages: Makn Sens a #twitter. In: *Proc. of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, (2011), pp. 368–378.

[3] Inouye, D., Kalita, J.: Comparing Twitter Summarization Algorithms for Multiple Post Summaries. In: *IEEE Third International Conference on Privacy, Security, Risk and Trust (PASSAT)*, (2011), pp. 298–306.

[4] Kamath, K., Caverlee, J.: Expert-Driven Topical Classification of Short Message Streams. In: *IEEE third international conference on Privacy, security, risk and trust (PASSAT)*, (2011), pp. 388–393.

[5] Kang, R., Fu W., Kannampallil, T.: Exploiting knowledge-in-the-head and knowledge-in-the-social-web: effects of domain expertise on exploratory search in individual and social search environments. In: *Proc. of the SIGCHI Conference on Human Factors in Computing Systems*, (2010), pp. 939–402.

[6] Nagmoti, R., Teredesai, A., De Cock, M.: Ranking Approaches for Microblog Search. In: *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, (2010), pp. 153–157.

[7] Nielsen, F.: A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. In: *Proc. of the ESWC2011 Workshop on 'Making Sense of Microposts': Big things come in small packages 718 in CEUR Workshop Proceedings*, (2011), pp. 93–98.

[8] O'Connor, B., Krieger, M., Ahn, D.: TweetMotif: Exploratory Search and Topic Summarization for Twitter. In: *Proc. of the Fourth International AAAI Conference on Weblogs and Social Media*, (2010), pp. 384–385.

[9] Uysal, I., Croft, W.: User Oriented Tweet Ranking: A Filtering Approach to Microblogs. In: *Proc. of the 20th ACM international conference on Information and knowledge management*, (2011), pp. 2261–2264.

[10] Weng, J., Lim, E., Jiang, J. et al.: TwitterRank: Finding Topic-sensitive Influential Twitterers. In: *Proc. of the third ACM international conference on Web search and data mining*, (2010), pp. 261–270.

[11] Yajuan, D., Long, J., Tao, Q. et al.: An Empirical Study on Learning to Rank of Tweets. In: *Proc. of the 23rd International Conference on Computational Linguistics*, (2010), pp. 295–303.

Multiple Sources of Search Context, their Influence and Applicability

Tomáš KRAMÁR*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
kramar@fiit.stuba.sk

Abstract. Search context represents an important aid in fulfilling the hidden intent behind the search query. Many sources of search context have been recognized and researched, but their mutual interactions and applicability have not been studied yet. In this work we analyze three sources of search context: social-based, activity-based and seasonality-based. We introduce a context model and define a context algebra that allows easy merging of multiple contexts. We analyze applicability of these sources with respect to the query features and user features and study their mutual interactions.

A paper based in part on this paper was accepted for publication in 36th European Conference on Information Retrieval (ECIR 2014).

* Doctoral degree study programmer in field: Software Engineering
Supervisor: Professor Mária Bieliková, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

Exploratory Search in Digital Libraries Using Automatic Text Summaries

Róbert MÓRO*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
moro@fiit.stuba.sk

Abstract. Information seeking tasks are demanding especially for researching new domains. The difficulty lies in impossibility of precise specification of information needs together with the amount of information currently available on the Web. We focus in this paper on online digital libraries domain. We propose a method of navigation using automatic text summaries of documents, the keywords of which serve as navigation leads to a set of relevant documents. Users can choose their own leads to filter the search results and follow potentially useful leads added by other searchers. We evaluate our approach on the scenario of a researcher novice such as a young master or doctoral student. We realized our method in the bookmarking system Annota and provide an evaluation on the dataset of the ACM Digital Library.

A paper based in part on this paper was accepted for publication in Bulletin of IEEE Technical Committee on Digital Libraries, 2013 TPDL Doctoral Consortium Issue.

* Doctoral degree study programme in field: Software Engineering
Supervisor: Professor Mária Bieliková, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

Social Insect Inspired Approach for Visualization and Tracking of Stories on the Web

Štefan SABO*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
sabo@fiit.stuba.sk

Abstract. In this paper we present an approach to visualization of currently unfolding news stories extracted from the news articles published on the Web. Our approach utilizes a set of social insect inspired agents to construct a graph representation of the news article space, that aims to model the structure of the article space in a comprehensive manner. This allows for a dynamic approach that reflects the changes in article, thus tracking the news stories as new events unfold. To enhance the readability of our visual model, we overlay the original graph with a layer containing information about the most popular terms related to tracked news stories and we color-code the graph according to the recentness of extracted articles, which gives us a comprehensive means of reviewing the current news stories as they unfold.

A paper based in part on this paper was published in 5th World Congres on Nature and Biologically Inspired Computing (NaBIC 2013), IEEE, pp. 226-231.

* Doctoral degree study programme in field: Software Engineering
Supervisor: Professor Pavol Návrat, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

A Study on Influence of Students' Personal Characteristics on Collaborative Learning

Ivan SRBA*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
srba@fiit.stuba.sk

Abstract. Current research in the domain of computer supported collaborative learning is influenced by new methodologies which shift a part of responsibility for learning process from pedagogues to students. This idea seems very promising and it is underpinned by several educational theories. However, would it be possible to apply these methodologies also in our educational conditions? We discuss this issue based on employing educational data mining methods to analyse how different academic, personal and collaborative characteristics influence students' behaviour in a web-based educational systems.

A paper based in part on this paper was submitted to the European Journal of Psychology of Education, Springer.

* Doctoral degree study programme in field: Software Engineering
Supervisor: Professor Mária Bieliková, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

Using Site Specificity to Build Better User Model from Web Browsing History

Márius ŠAJGALÍK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
sajgalik@fiit.stuba.sk

Abstract. Extracting user interests from user browsing history has been already studied by several researchers. We present a novel method to enhance the quality of user interest extraction by harnessing the site specificity. We follow the idea that some sites are more generic and not so relevant to real user interests, whereas more specific sites are more relevant to the user interests. We compute the site specificity by analysing concept diversity in the site concept graph.

1 Introduction

The ever growing Web content has enabled users to stay just on the Web more often and accommodate their various needs. Users browse the Web to read news, work, play games, socialise, etc. Even news can cover more than one broader topic like politics, sport, or profession news. From the developer's point of view who endeavours to build the best possible general user model (assuming not to focus on a particular domain), not each of these are necessarily equally influent. As users browse the websites, they can have overlooked, or just skipped some parts they are not interested in. If we are to model the user interests, we often cannot assume that user has seen the whole page, nor even think it interested her. This is particularly applicable to websites of general interests like the aforementioned news portals. If there are multiple topics within a single website, it is highly probable that user is just not interested in all of these topics, but chooses to read only a few of them.

To account for these topic varieties, we decided to evaluate the specificity of each site and thus make it less influential on user model. In order to infer an interest of the user in a site, we propose to calculate the site specificity. The less tightly related topics are contained within a site, the more specific it is and the more probable is the higher significance of the discovered topics for user interests. On the other hand if there are multiple topics, yet vaguely related if at all, the probability of user interest in all of them is very low.

To be able to discover some measurable features, which might influence the overall site specificity, we need some additional knowledge about the web content. However, there is still not enough explicit semantic information of sufficient quality included in the webpage content, which

* Doctoral degree study programme in field: Software Engineering
Supervisors: Dr. Michal Barla, Professor Mária Bieliková, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

forces us to incorporate some kind of ontology to understand the content of the "wild web" better [2]. Therefore, we use WordNet, which can be considered as a form of a lightweight ontology.

2 Related Works

Our proposed method of the site specificity identification builds mainly upon *Text Representation with WordNet Synsets using Soft Sense Disambiguation* [10]. There, instead of words, the text is represented by WordNet synsets. Its author points out two major drawbacks of "bag of words" representation – polysemy and synonymy. Polysemy cause the ambiguity of the words since a single word can have multiple meanings. In case of synonymy, several words can have the same meaning and bag of words just lack the information about such relations among them. On the other hand, in synset representation, these synsets stand for concepts corresponding to the words in the text. Evaluation in [10] shows that the best way to rank synsets in order to infer the most probable word meaning (referred to as the soft sense disambiguation) is achieved by using the Page rank algorithm.

Use of the page ranking algorithm in text mining has already been researched. One of the pioneer methods is TextRank [6]. It creates a graph, where words are vertices and edges connect word collocations. Its goal is to extract keywords, which importance is propagated via those collocation links. Its promising results even caused its authors to apply for a patent on it.

Author in [4] uses the aforementioned synset representation to extract document summarisation. It is a bit similar to our approach beyond just the synset representation since we use some similar basic pre-processing techniques like POS tagging. In addition, according to the ranked synsets, it ranks the sentences whilst calculating the similarity among them, so that only the top five sentences with the least semantic similarity are extracted.

2.1 Concept similarity measures

There are many approaches to compute the semantic similarity between ontological concepts, but basically they can be divided into three categories:

- node-based approaches,
- edge-based approaches,
- hybrid approaches.

Node-based approaches are based on considering concept weights whereas edge-based methods consider semantic relations among concepts to calculate their similarity. Hybrid combine different resources to calculate the estimated semantic similarity.

WordNet::Similarity [9] is a Perl module focused on implementing variety of semantic similarity and relatedness measures based on information found in WordNet lexical database. In [8] we can find an empirical comparison of results achieved of the three widely used similarity measures for pairs of concepts based on the information content.

The information content is a measure of specificity for a concept. The higher value of information content, the more specific is the concept (e.g. violin). On the other hand, lower values signify more general concepts (e.g. object). The information content *IC(c)* of a concept *c* is defined as the negative logarithm of the probability of this concept:

$$IC(c) = -\log P(c) \tag{1}$$

The probability of a concept *P(c)* is computed as relative frequency of it:

$$P(c) = \frac{freq(c)}{N} \tag{2}$$

N is the total number of nouns observed in a text corpus (and present in WordNet corpus as well) and *freq(c)* denotes the concept frequency:

$$freq(c) = \sum_{n \in words(c)} count(n) \quad (3)$$

There, *words(c)* is the set of words assigned to the WordNet concept *c* and *count(n)* is the total number of occurrences of the noun *n*.

As mentioned in [8], there are three widely used measures based on information content. All these measures are based on the Resnik measure [11], which uses the notion of a least common subsumer (LCS). LCS of two concepts c_1 and c_2 is the most specific concept reachable from both of them. The Resnik similarity measure is calculated as the information content of the LCS:

$$res(c_1, c_2) = \mathrm{IC}(\mathrm{LCS}(c_1, c_2)) \quad (4)$$

However, sometimes it is not unambiguous to calculate the information content of LCS, since there can exist multiple different LCSs. We can notice one of these ambiguous cases in the fragment of WordNet hierarchy in Figure 1.

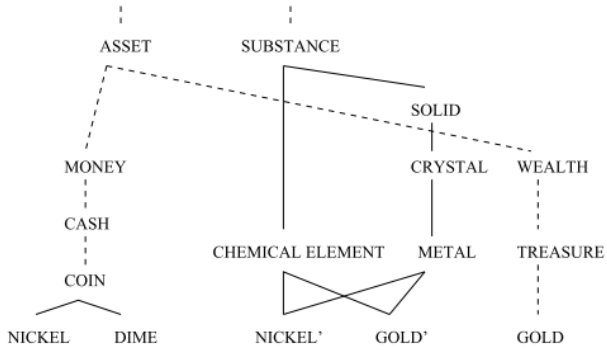


Figure 1. A fragment of the WordNet hierarchy, where NICKEL and GOLD have ambiguous LCS.

To account for this ambiguity, Resnik [11] provides an easy and intuitive solution – we simply take the maximum of information contents of all LCSs, since that represent the tightest semantic relation between given concepts. The other two measures are related to Resnik measure, as they both use it in their calculations. They attempt to get better results by counting the information content also of the individual concepts. We had attempted to use them in our experiments, however, they did not prove to be of any positive contribution to the quality of our results.

3 Website Specificity

The basic idea of our approach is to compute the specificity of just a single webpage. To generalise it to an arbitrary set of webpages, we simply concatenate them and calculate the specificity over the union. To compute the website specificity, we choose several subpages within it and concatenate their content into a single piece of text. As we are focusing on enhancing a user model within the web browser, we choose only those subpages of the website, which are present in user's browsing history.

At first, we extract an article and choose only the noun terms as keyword candidates. With these feasible words extracted, we take all the noun synsets of WordNet, which contain at least one of these feasible terms. We call these synsets the basis synsets. Then we create the concept graph *G = (V, E)*, where vertices *V* are all the basis synsets plus those reachable by following hypernym or holonym relations. This aims to influence also the more general concepts (WordNet synsets) to get to the broader topics discussed in the extracted article. Though we call *G* the concept graph, it is proper to note that not all of the WordNet synsets represent concepts. Some of them represent

instances of concepts, however, it is not significant for correctness of this method, since we are interested mainly in the most general topics contained in a text. More detailed discussion on this difference between concepts and concept instances can be found in [1].

After we have built this concept graph, we perform page ranking algorithm to infer the relevance of individual concepts. This algorithm is inspired by Google's Page rank [3]. However, our version of page ranking is adapted from approach in [10], where the author notes that the in-degree alone is not a sufficient indicator of the concept authority. Therefore we consider also the out-degree of the synset nodes.

The principle of our approach is to do a two-pass ranking. In the first pass, we propagate the authority of a synset to all hypernyms and holonyms to obtain the most probable word senses. Apart from the method presented in [10], we consider the information content of single concepts. That is, we multiply the vertex score obtained from the Page rank algorithm by the information content of the corresponding concept. Besides these hierarchy links, we consider collocations too. That is, after performing the page ranking in the first pass, we link also the neighbouring terms in the second pass as well to support the collocated word senses and thus, get the key concepts. We adapted this idea from TextRank [6], which is an unsupervised method to extract keywords. The inference is done iteratively using this formula to compute a new vertex score:

$$VS(V_i) = (1-d) + d * \sum_{V_j \in In(V_i)} \frac{w_{ji}}{\sum_{V_k \in Out(V_j)} w_{jk}} VS(V_j) \quad (5)$$

There, *VS(v)* denotes the vertex score of vertex *v*, *In(v)* denotes the set of all incoming edges and *Out(v)* denotes the set of all outgoing edges. Edge between vertices V_i and V_j is denoted as w_{ij} and d is the damping factor usually set to 0.85 [3, 6].

3.1 Specificity measures

To calculate the site specificity, we apply various measures of concept similarity to measure the topic diversity or more specifically, the semantic coherence of the topmost concepts contained in the concept graph. We do not consider all concepts in the graph, but only those ranked at the top after inferring the ranking algorithm, since those are the most probable to be the most relevant representatives of the covered topics. These topmost concepts should cover all major topics contained in the text within the website. If there is only one topic covered by the text, we hypothesise that these top concepts should be more strongly semantically connected among themselves. On contrary, if the text contains more than one topic, some of these top concepts should be less strongly semantically connected than in the previous case. We propose four possible measures to measure this semantic coherence:

- the average similarity among all pairs of the topmost concepts,
- the lowest similarity among all pairs of the topmost concepts,
- the information content of the subsumer of the topmost concepts,
- the last cluster-merging edge.

The first and second approach are rather self-explaining. In the first case, we simply take the average of similarity values between every pair of the topmost concepts. In the second case, we take just the minimum of these pairwise similarities.

The third measure can be seen as a generalisation of the Resnik similarity measure. Unlike the Resnik measure, we do not consider just two concepts, but we want to evaluate the information content among all topmost concepts. This requires finding the least common subsumer (LCS) of all these topmost concepts. We can see (also depicted in Figure 1) that some concepts can have multiple hypernyms, which makes the search a little complicated also in the simple pairwise Resnik similarity computation. To be able to find the LCS of multiple concepts quickly (there can be also several of them according to Figure 1), we devised the following algorithm to do so:

```
1. Let V be an array, where V[i] is visit count for concept i
2. Initialise all V[i] = 0
3. Let LCS = [0, 0]
4. Let C be an array of all topmost concepts
5. Iterate over C, where C[i] is the i-th concept (zero-based index)
   5.1. Let Q be an empty queue
   5.2. Insert C[i] into Q
   5.3. While Q is non-empty
     5.3.1. Remove the first element c from Q, where c is a concept
     5.3.2. If V[c] = i
            5.3.2.1. If LCS[0] = i or LCS[1] < IC(c)
                     5.3.2.1.1. Let LCS = [i + 1, IC(c)]
                     5.3.2.1.2. Increment V[c]
     5.3.3. For each hypernym h of c
            5.3.3.1. Insert (h,d+1) into Q
6. If LCS[0] is lower than number topmost concepts
   6.1. Return the website specificity = 0
7. Else return the website specificity = LCS[1]
```

After performing this algorithm, we get the final website specificity. The `IC(c)` stands for the value of information content (see Formula 1) of concept c.

The fourth proposed way to measure the semantic coherence of the topmost concepts is based on clustering. We attempt to cluster the topmost concepts into topics until we connect all of them. Finally, we evaluate the site specificity as the semantic similarity between the last connected concepts. The algorithm goes as follows:

```
1. Assign each top k concept to a separate cluster
2. Set the global semantic coherence GSC = 0
3. Sort all edges between pairs of top k concepts in order of their
   similarity
4. Iterate through sorted edges
   a. Let v1 and v2 be the pair of vertices (concepts) connected by
      the current edge e
   b. If v1 and v2 belong to the same cluster, continue with the
      step 4
   c. Merge clusters containing v1 and v2
   d. If there are more than one cluster remaining, continue with
      the step 4
   e. Let s be the semantic similarity between concepts v1 and v2
   f. Set GSC = s
   g. We have found the result (GSC), quit
```

As we can see, this algorithm is pretty similar to Kruskal algorithm to compute the minimum spanning tree and thus, it can be implemented efficiently using the disjoint-set data structure in $O(n^2 \log n)$ time, where n is the number of topmost concepts.

From another perspective, it can be considered as an instance of the agglomerative hierarchical clustering method [5], since the clustering is dependent solely on the similarity measures (the initial sorting) and we are using the "bottom up" approach. We also use the idea of a hypothetical root concept to subsume all the concepts.

This allows us to apply the similarity measures to any pair of concepts and guarantees that after we had iterated through all the edges, we have found the desired result stored in GSC, which denotes the global semantic coherence of the concept graph partition containing the top concepts.

4 Evaluating the Site Specificity

At first, we present the promising results of our key-concept extraction algorithm, which makes the basis for evaluating the site specificity. In Table 1, we present an example of the sample results of our key-concept extraction method. To showcase our contribution in the proposed concept ranking method, we compare two distinct approaches. In the first approach, we show the results of the mere page ranking of the concepts using just the hypernymy, holonymy and collocation relations among them. The second approach considers also the information content of the concepts, as we have already described.

Table 1. Extracted key-concepts from the Wikipedia page about the data structures.

URL address: http://en.wikipedia.org/wiki/Data_structure	
Not considering information content	Considering information content
– data, information – union, labor union, trade union, trades union, brotherhood – memory, computer memory, storage, computer storage, store, memory board – phonograph record, phonograph recording, record, disk, disc, platter – structure, construction – type – library – order – hashish, hasheesh, haschisch, hash – phylum	– data, information – type – array – structure, construction – computer memory unit – record – memory, computer memory, storage, computer storage, store, memory board – class – model, example – queue

We can see that using the second approach (the right part of Table 1), the results are more reasonable compared to the more noisy concepts produced by the first approach (the left part of Table 1). At the second thought, we can observe a quite intuitive explanation, if we put it into an analogy with the common used TF-IDF method. By performing just a page ranking of the first approach, we are calculating only the TF-like factor. Similarly, the notion of the information content of a concept is pretty analogical to the inverse document frequency part of TF-IDF method.

After we have succeeded to extract such relevant and rather noiseless key-concepts, we are able to measure similarities between them. Our hypothesis is that the websites, which contain more semantically related concepts are more specific. We evaluated the site specificity using all four proposed measures for manually chosen sample websites. These belong to some internationally popular websites like Wikipedia and various news portals. To evaluate the correctness of our approach, we came up with simple solution on how to generate both more specific and less specific representatives. To get less specific websites, we consider some generic sites like Wikipedia or New York Times news portal as a whole. To account for more specific websites, we simply consider just a single webpages within those generic websites. Thus, we can easily compare them to evaluate the correctness of our solution. Table 2 summarises these results. For each website (webpage), there are four values corresponding to the calculated specificity score of each proposed measure (from left to right, there is a score for the first to the fourth measure) and we also present the extracted key-concepts, which form the basis for all these measures as they are the very object being measured.

Table 2. Site specificity scores and extracted key-concepts for different websites.

http://en.wikipedia.org/	3.991	0	0	6.024
World Wide Web, WWW, web | album | stereo, stereoscopic picture, stereoscopic photograph | billboard, hoarding | music | news | Earth, earth, world, globe | china | view, aspect, prospect, scene, vista, panorama | game				
http://en.wikipedia.org/wiki/Data_structure	5.1	2.773	6.024	6.024
data, information | type | array | structure, construction | computer memory unit | record | memory, computer memory, storage, computer storage, store, memory board | class | model, example | queue				
http://en.wikipedia.org/wiki/Disjoint-set_data_structure	5.253	3.276	6.024	6.024
list, listing | lymph node, lymph gland, node | union | root, tooth root | parent | tree | data, information | routine, subroutine, subprogram, procedure, function | structure, construction | component, constituent, element, factor, ingredient				
http://en.wikipedia.org/wiki/Latent_Dirichlet_allocation	4.523	2.894	6.024	6.024
subject, topic, theme | model, simulation | distribution | word | actor's line, speech, words | document, written document, papers | metric weight unit, weight unit | derivation | probability | vector				
http://en.wikipedia.org/wiki/South_Dakota	5.149	1.201	6.024	6.024
south, due south, southward, S | World Wide Web, WWW, web | state | population | Indian, American Indian, Red Indian | Black, Black person, blackamoor, Negro, Negroid | web | class | census, nose count, nosecount | waterfall, falls				
http://en.wikipedia.org/wiki/Politics	3.279	0	0	4.22
politics | citation | state | government, authorities, regime | law | putrescence, putridness, rottenness, corruption | power, powerfulness | administration, disposal | institution, establishment | Earth, earth, world, globe				
http://www.nytimes.com/	4.088	0	0	6.024
pope, Catholic Pope, Roman Catholic Pope, pontiff, Holy Father, Vicar of Christ, Bishop of Rome | first base, first | church, Christian church | Earth, earth, world, globe | euro | class | school | benedick, benedict | keyboard | device				
http://www.nytimes.com/2013/03/17/sunday-review/reading-writing-and-video-games.html	5.123	2.773	6.024	6.024
school | class | classroom, schoolroom | mathematics, math, maths | microwave | genus | technology, engineering | apple | video, picture | rung, round, stave				
http://www.nytimes.com/reuters/2013/03/16/sports/soccer/16reuters-soccer-italy.html	5.135	2.077	6.024	6.024
bologna, Bologna sausage | sequin, spangle, diamante | shot, pellet | athletic game | minutes, proceedings, transactions | rung, round, stave | minute, arcminute, minute of arc | game | testis, testicle, orchis, ball, ballock, bollock, nut, egg | chair				

As we can see, the results are promising, though there is not a single winner among the proposed measures. Measure 4 seems to be the least significant of all four measures. On the contrary, the measure 1 seems to be the most significant. We can also see an interesting phenomenon in results of measure 1 for the New York Times website, where we can observe the temporal gain in specificity due to the global interest in the new Pope election. This shows the importance of a time variable. We can observe that measure 2 and 3 correctly distinguish whether the site is multi-topical or not. Measure 3 is even behaving like a binary variable, since if there are multiple topics, the score is zero, and 6.024 otherwise. There are several possible enhancements that could positively influence the quality of results. We have chosen to take only the top 10 concepts. This decision may be not quite optimal for each measure. Also the page ranking in the concept graph has lots of tuneable parameters and we may be not using the right combination of them.

5 Conclusions

In this paper we have presented a novel notion of the site specificity to account for different interests in different sites based on their topic diversity. We proposed four different methods and evaluated them using four different measures. The presented results could be enhanced further by constructing a probabilistic model (Bayesian network) based on different features of the constructed concept graph. As we can see in the evaluation, each of proposed measures has different influence on the site specificity. Thus, we could train the model parameters corresponding to different features using some dataset of categorised websites. Using such model, we would be able to set the values of observed variables and do the inference to obtain the conditional probability of the website being specific. We also plan to use the results presented in this paper to devise another method, which we believe will build a better user model having taken the website specificity into account.

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/0675/11.

References

[1] Alfonseca, E., Manandhar, S.: Distinguishing concepts and instances in WordNet. In: *Proc. of the first Int. Conf. of Global WordNet Association*. Mysore, India, (2002).

[2] Bieliková, M., Barla, M., Šimko, M.: Lightweight Semantics for the "Wild Web". Keynote. In: *WWW/Internet 2011, Proc. of the IADIS Int. Conf.*, IADIS Press, (2011), pp. xxv–xxxii.

[3] Brin, S., Page, L.: The anatomy of a large-scale hyper-textual Web search engine. *Computer Networks and ISDN Systems*, Vol. 30, (1998), pp. 1–7.

[4] Dang, C., Luo, X.: WordNet-based Document Summarization. In: *Proc. of 7th WSEAS Int. Conf.*, (2008), pp. 383–387.

[5] Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning. *Springer Series in Data Mining, Inference and Prediction*. (2011), pp. 520–526.

[6] Mihalcea, R., Tarau, P.: TextRank: Bringing Order into Texts. In *Conf. on Empirical Methods in Natural Language Processing* (2004), pp. 404–411.

[7] Miller, G. A.: WordNet: A Lexical Database for English. *Communications of the ACM*, Vol. 38, No. 11, (1995), pp. 39–41.

[8] Pedersen, T.: Information Content Measures of Semantic Similarity Perform Better Without Sense-Tagged Text. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (HLT '10)*, Stroudsburg, PA, USA, (2010), pp. 329–332.

[9] Pedersen, T., Patwardhan, S., Michelizzi, J.: *WordNet:* Similarity: measuring the relatedness of concepts. In *Demonstration Papers at HLT-NAACL 2004*. Association for Computational Linguistics, Stroudsburg, PA, USA, (2004), pp. 38–41.

[10] Ramakrishnanan, G., Bhattacharyya, P.: Text Representation with WordNet Synsets Using Soft Sense Disambiguation. *Ingénierie des systèmes d'information*, vol. 8, (2003), pp. 55–70.

[11] Resnik, P.: Using Information Content to Evaluate Semantic Similarity in a Taxonomy. In: *Proc. of the 14th int. joint conf. on Artificial intelligence (IJCAI'95)*, Chris S. Mellish (Ed.), Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, vol. 1, (1995), pp. 448–453.

Crowdsourcing in the Class

Jakub ŠIMKO*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
jsimko@fiit.stuba.sk

Abstract. The goal of this research is to involve and examine a crowd-based metadata acquisition approach within an online learning framework. The task is to validate free text answers to questions related to a course through crowd of students. We build on our previous research, where we have shown, that the aggregate student crowd answer can be correct to some extent. Nevertheless, it is a challenge to extract more accurate crowd answer from the individual student answers. On this, we focus in our work. Our aim is to measure and exploit information about student expertise level (for the course domain) by marginalizing the answers of "bad" students and strengthening answers of "good" students.

A paper based in part on this paper was published in 5th Int. Conference on Computational Collective Intelligence Technologies and Applications (ICCC 2013), Springer, pp. 62-71.

* Doctoral degree study programme in field: Software Engineering
Supervisor: Professor Mária Bieliková, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

Computer Graphics and Computer Vision

Planar Object Recognition in an Augmented Reality Application on Mobile Devices

Marek JAKAB*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
marko.jakab@gmail.com

Abstract. The purpose of our research is to develop an application of augmented reality on mobile device, which will be educative and entertaining for their users – children. User will be asked for an input to take a picture from the book and the application will draw a supplementary information in the form of a 3D object on the screen. The key task of our application is the problem of image recognition on mobile platform using local descriptors. Currently available descriptors included in OpenCV library are well designed, some of them are scale and rotation invariant, but most of them are time and memory consuming and hence not suitable for mobile platform. Therefore we decided to develop a fast binary descriptor based on the Histogram of Intensity PatcheS (HIPs) originally proposed by Simon Taylor et al. To train the descriptor, we need a set of images derived from a reference picture taken under varying viewing conditions and varying geometry. Our descriptor is based on a histogram of intensity of the selected pixels around the key-point in such a way that rotation of the patches could be implemented very efficient in the form of buffer shift. We use this descriptor in the combination with the FAST key-point detector and a sophisticated method of key-points selection is then used with the aim to reduce the computation time.

Amended version published in Information Sciences and Technologies Bulletin of the ACM Slovakia, Vol. 5, No. 2 (2013), pp. 27-31.

* Bachelor degree study programme in field: Informatics
Supervisor: Dr. Vanda Benešová, Institute of Applied Informatics, Faculty of Informatics and Information Technologies STU in Bratislava

Eye Blink Detection

Patrik POLATSEK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
patrik.polatsek@gmail.com

Abstract. Nowadays, people spend more time in front of electronic screens like computers, laptops, TV screens, mobile phones or tablets which cause eye blink frequency to decrease. Each blink spreads the tears on the eye cornea to moisture and disinfect the eye. Reduced blink rate causes eye redness and dryness also known as Dry Eye, which belongs to the major symptoms of the Computer Vision Syndrome. The goal of this work is to design eye blink detector which can be used in dry eye prevention system. We have analyzed available techniques for blink detection and designed our own solutions based on histogram backprojection and optical flow methods. We have tested our algorithms on different datasets under various lighting conditions. Inner movement detection method based on optical flow performs better than the histogram based ones. We achieve higher recognition rate and much lower false positive rate than the-state-of-the-art technique presented by Divjak and Bischof.

1 Introduction

The number of people using computers every day increases. There are also more people who suffer from symptoms collectively called *Computer Vision Syndrome* (CVS). It is a set of problems related to computer use. The rate of unconscious eye blinking while looking at luminous objects within close distance reduces significantly (up to 60 % reduction). Blinking helps us to spread the tear film and moisten the surface of the eye, due to which the reduced rate of blinking leads to *Dry Eye*. Typical ocular complaints experienced by intensive computing work (more than 3 hours per day) include dryness, redness, burning, sandy-gritty eye irritation or sensitivity to light and eye fatigue. The easiest way to avoid the symptoms of Dry Eye is to blink regularly [3, 13].

Our aim is to create eye blink detector, which could be used in real-time blink detection system. In case of low blink rate it will notify a user to blink more frequently. This paper proposes two different methods on blink detection. The first of presented algorithms computes *backprojection* from 1D saturation and 2D hue-saturation histogram. The second method addressed as *Inner movement detection* detects eyelid motion using *Lucas-Kanade* (KLT) feature tracker [11].

* Bachelor study programme in field: Informatics
Supervisor: Andrej Fogelton, Institute of Applied Informatics, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 211–216.

2 Related Work

Optical flow in [7] tracks eyelid movements to detect eye blinks. Detection is based on matching SIFT (scale-invariant feature transform) descriptors computed on GPU. First, thresholded frame difference inside the eye region locates motion regions. Consequently, these regions are being used to calculate the optical flow. While user blinks, eyelids move up and down and the dominant motion is in vertical direction. This method detects 97% of blinks on their dataset. Most of the false positive detections are the result of gaze lowering and vertical head movements. Method based on optical flow estimation is also presented in [4]. It locates eyes and face position by 3 different classifiers. The algorithm is successful mostly when the head is directly facing the camera. The KLT tracker is used to track the detected feature points. This blink detector uses GPU-based optical flow in the face region. The flow within eyes is compensated for the global face movement, normalized and corrected in rotation when eyes are in non-horizontal position. Afterwards dominant orientation of the flow is estimated. The flow data are processed by adaptive threshold to detect eye blinks. Authors report good blink detection rate (more than 90%). However this approach has problems with detecting blinks when eyes are quickly moving up and down.

The eyelid movements are estimated by *normal flow* instead of optical flow in [6]. It is the component of optical flow that is orthogonal to the image gradient. Authors claim that the computation of normal flow is more effective than the previous method.

Arai et al. present Gabor filter-based method for blink detection in [1]. *Gabor filter* is a linear filter for extracting contours within the eye. After applying the filter, the distance between detected top and bottom arc in eye region is measured. Different distance indicates closed or opened eye. The problem of arc extraction arises while the person is looking down.

Variance map specifies distribution of intensities from the mean value in an image sequence. The intensity of pixels located in eye region changes during the blink, which can be used in detection process as in [10].

Correlation measures the similarity between actual eye and open eye image. As someone closes eyes during the blink, correlation coefficient decreases. Blink detection via correlation for immobile people is presented in [5].

A blink detection algorithm in [9] is based on the fact that the *upper* and *lower part* of eye have different distribution of *mean intensities* during open eyes and blinks. These intensities cross during the eyelid closing and opening.

Liting et al. [8] use a deformable model - *Active Shape Model* represented by several landmarks as the eye contour shape. Model learns the appearance around each landmark and fits it in the actual frame to obtain new eye shape. Blinks are detected by the distance measurement between upper and lower eyelid.

Ayudhaya et al. [2] detect blinks by the *eyelid's state detecting* (ESD) *value* calculation. It increases the threshold until the resulting image has at least one black pixel after applying median blur filtering. This threshold value (ESD) differs while user blinks.

3 Proposed Algorithms

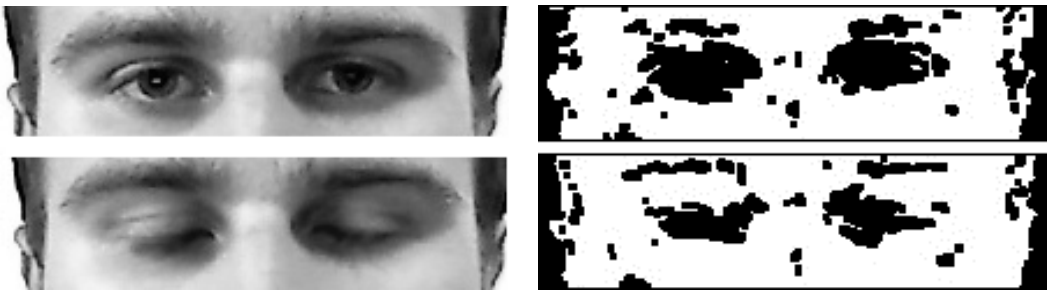
Due to our goal to do CVS preventing system, our main focus is on model situation when the user is facing the computer screen. Because of this, the high recognition rate within the efficient computation is necessary.

We introduce two methods based on histogram backprojection and the Inner movement detection based on KLT feature tracker.

3.1 Histogram Backprojection

We use *histogram* to represent skin color of the user. *Histogram backprojection* creates a probability map over the image. In other words backprojection determines how well the pixels from the image

fit the distribution of a given histogram. Higher value in a backprojected image denotes more likely location of the given object. We detect closed eyes as high percentage of skin color pixels within the eye region otherwise we consider eyes opened (Figure 1).



(a) Source images. (b) Backprojected images.

Figure 1. Histogram backprojection for a person with open and closed eyes.

We use the *HSV* (Hue Saturation Value) color model to achieve partial luminance invariance by the omission of the Value channel. We did experiments with two different histograms:

- 1D *saturation* histogram (histogram S),
- 2D *hue-saturation* histogram (histogram HS).

First we detect the user's face by *Haar Cascade Classifier* [12]. We calculate the skin color histogram from a sequence of images of face regions. Other parts of the image are not used to obtain as precise skin color histogram as possible. Histogram is normalized afterwards and regularly updated. For every input image we calculate the backprojection with this histogram. Subsequently a resulting backprojected image is modified using morphological operations (Open and Erode) and threshold ($threshold_{HS} = 10$ in hue-saturation and $threshold_S = 25$ in saturation histogram obtained by experiments) to increase small difference between open and closed eyelids due to lower skin probability of eyelids caused by shadows in eye areas or make-up. Finally the average value of the probabilities is calculated from the region of the user's eyes. Significant increase is considered as eye blink of the user. Figures 3(a) and 3(b) illustrate results of backprojection while using different histograms.

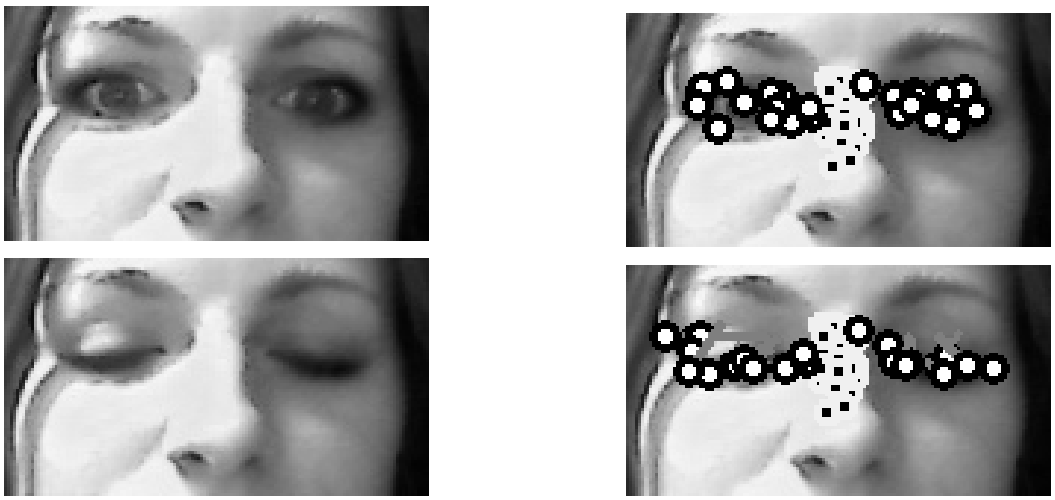
3.2 Inner Movement Detection

We introduce our own *Inner Movement Detection* algorithm based on *optical flow*. Optical flow locates new feature position in the following frame. One of the most common method called KLT tracker [11] selects features suitable for tracking with high intensity changes in both directions.

If a user blinks, the mean displacement of feature points within the eye region should be greater than the displacement of the rest of the points within the face area (Figure 2).

The first step consists of localizing a user's face and eyes using *Haar Cascade Classifier* [12] on grayscale image. We initialize random KLT features within the eye and nose regions and classify them as left ocular, right ocular or non-ocular. These features are being tracked by KLT tracker. Tracking is reinitialized in regular intervals or in case of loss of many feature points. We compute the average displacement separately for three groups of points. Afterwards we compare the difference between the left or right ocular and non-ocular movement displacements. If this difference exceeds a threshold value ($threshold_{diff} = face.height/165$, where $face.height$ is the height of detected face in the initial phase), a movement within the eye region is anticipated. Consequently we count the ratio of ocular points that moved down at least of a specific

distance in the direction of y-axis ($distance_y = face.height/110$) in order to exclude false positives caused by horizontal eye movements. Due to proper computation of the ratio we eliminate the vertical ocular displacement caused by head movements. The ocular points are therefore shifted in a distance equal to the average displacement of non-ocular points. If the ratio is higher than a threshold (5% of displacements of one group of ocular points and 2% of displacement of the second one), we consider it as a blink. Figure 3(c) represents a graph of values defined as $max(abs(avg(left) - avg(non)), abs(avg(right) - avg(non)))$, where max and abs are the maximum and absolute value, avg indicates the average movement within a given region and $left$, $right$ and non denote left ocular, right ocular and non-ocular region.



(a) Source images. (b) Displacement of feature points.

Figure 2. Tracking of feature points using KLT feature tracker by eye blink.

4 Evaluation

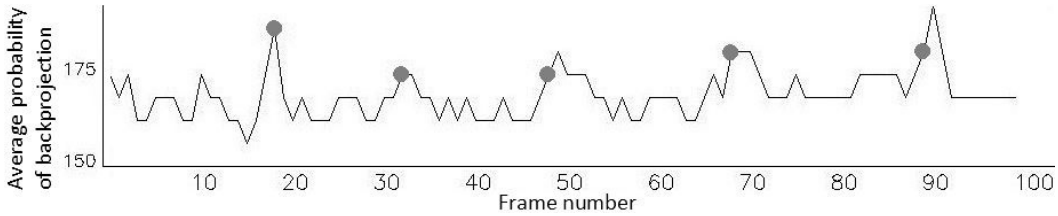
Our blink detection algorithms are evaluated on two datasets. Our own dataset includes 8 individuals (5 males and 3 females, one person wearing glasses) under different lighting conditions who sit in front of a computer screen mostly in a stable position and looking directly at the screen. It consists of 7569 frames and 128 blinks. The second image sequence - *the Talking Face Video* (TALKING) is publicly available from Inria[1]. It includes 5000 images of a person engaged in conversation who blinks 61 times.

We have tested our algorithms and compared their blink detection abilities to the optical flow method mentioned in [4]. The results are shown in Table 4. The best true and false positive rate are achieved by Inner movement detection. It detects 93,75% of blinks on own dataset and 98,36% of blinks on the Talking Face Video.

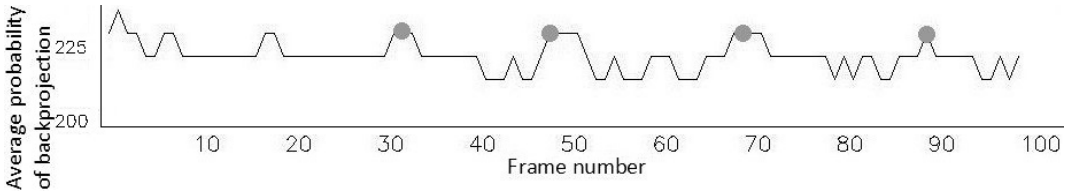
Backprojections using hue-saturation and saturation histogram provide similar accuracies. Values of saturation channel of an image differ in case of skin and pupil in most light conditions, thus it provides reliable information about user's blinks. However hue channel is often different in whole eye regions. Sometimes it is without any significant changes when eye blinks. It happens mostly in very dark images. False detections are the results of luminance changes, poor light conditions,

[1] `http://www-prima.inrialpes.fr/FGnet/data/01-TalkingFace/talking_face.html`

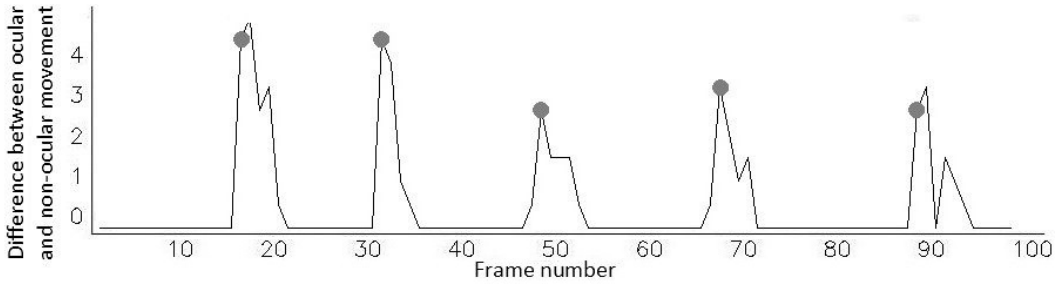
changes in gaze direction, facial mimicry such as smiling and eyelid makeup. In such cases it is very difficult to recognize whether a user blinks or not. Backprojection using hue-saturation histogram has many missed blinks when an individual wears glasses. Inner movement detection, our best method, has 14 false positive and 9 false negative cases caused mainly by rapid head movements, lowering the gaze and reflection from glasses.



(a) Graph of the average values of backprojection using 2D hue-saturation histogram.



(b) Graph of the average values of backprojection using using 1D saturation histogram.



(c) Graph of the differences between the average ocular and non-ocular movement within the face area.

Figure 3. Graphs produced using our eye blink detection algorithms. They are computed from an image sequence of a user while working in front of computer. The user blinks at frames 18, 33, 50, 69 and 90. Detected blinks are represented by circles on the graph.

Table 1. Comparison of our blink detection algorithms to the method in [4]. TP represents true positive rate and FP is false positive rate.

Method	**Own dataset**		**TALKING**	
	TP	*FP*	*TP*	*FP*
Backprojection (Histogram S)	81,25%	0,40%	88,52%	0,49%
Backprojection (Histogram HS)	75,00%	0,32%	85,25%	0,47%
Inner movement detection	93,75%	0,05%	98,36%	0,20%
Method in [4]	-	-	95%	19%

5 Conclusion

In this paper we proposed two techniques for eye blink detection. The first method detects blinks by backprojection using saturation or hue-saturation histogram. The second method is based on KLT feature tracker, which tracks eyelid motions. The model situation is a user looking at the computer screen. Inner movement detection method outperforms the method in [4]. It provides over 3% better true positive rate and about 18% lower false positive rate.

Acknowledgement: This project is supported by Tatra bank foundation E-Talent 2012et009.

References

[1] Arai, K., Mardiyanto, R.: Comparative Study on Blink Detection and Gaze Estimation Methods for HCI, in Particular, Gabor Filter Utilized Blink Detection Method. In: *Proceedings of the 2011 Eighth International Conference on Information Technology: New Generations*. ITNG '11, Washington, DC, USA, IEEE Computer Society, 2011, pp. 441–446.

[2] Ayudhaya, C., Srinark, T.: A method for a real time eye blink detection and its applications. In: *The 6th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, 2009, pp. 25 – 30.

[3] Blehm, C., Vishnu, S., Khattak, A., Mitra, S., Yee, R.W.: Computer Vision Syndrome: A Review. *Survey of Ophthalmology*, 2005, vol. 50, no. 3, pp. 253 – 262.

[4] Divjak, M., Bischof, H.: Eye blink based fatigue detection for prevention of Computer Vision Syndrome. In: *IAPR Conference on Machine Vision Applications (MVA 2009)*, 2009, pp. 350–353.

[5] Grauman, K., Betke, M., Gips, J., Bradski, G.: Communication via eye blinks - detection and duration analysis in real time. In: *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*. Volume 1., 2001, pp. I–1010 – I–1017 vol.1.

[6] Heishman, R., Duric, Z.: Using Image Flow to Detect Eye Blinks in Color Videos. In: *Applications of Computer Vision, 2007. WACV '07. IEEE Workshop on*, 2007, p. 52.

[7] Lalonde, M., Byrns, D., Gagnon, L., Teasdale, N., Laurendeau, D.: Real-time eye blink detection with GPU-based SIFT tracking. In: *Proceedings of the Fourth Canadian Conference on Computer and Robot Vision*. CRV '07, Washington, DC, USA, IEEE Computer Society, 2007, pp. 481–487.

[8] Liting, W., Xiaoqing, D., Changsong, L., Wang, K.: Eye Blink Detection Based on Eye Contour extraction. In: *Image Processing: Algorithms and Systems*, SPIE Electronics Imaging, 2009, p. 72450.

[9] Moriyama, T., Kanade, T., Cohn, J., Xiao, J., Ambadar, Z., Gao, J., Imanura, M.: Automatic recognition of eye blinking in spontaneously occurring behavior. In: *Proceedings of the 16th International Conference on Pattern Recognition (ICPR '2002)*. Volume 4., 2002, pp. 78 – 81.

[10] Morris, T., Blenkhorn, P., Zaidi, F.: Blink detection for real-time eye tracking. *J. Netw. Comput. Appl.*, 2002, vol. 25, no. 2, pp. 129–143.

[11] Tomasi, C., Kanade, T.: Detection and Tracking of Point Features Technical Report CMU-CS-91-132. *Image Rochester NY*, 1991, vol. 91, no. April, pp. 1–22.

[12] Viola, P.A., Jones, M.J.: Rapid Object Detection using a Boosted Cascade of Simple Features. In: *CVPR*, IEEE Computer Society, 2001, pp. 511–518.

[13] Yan, Z., Hu, L., Chen, H., Lu, F.: Computer Vision Syndrome: A widely spreading but largely unknown epidemic among computer users. *Computers in Human Behavior*, 2008, vol. 24, no. 5, pp. 2026 – 2042.

Superpixel Image Clustering

Andrej FOGELTON*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
fogelton@fiit.stuba.sk

Abstract. Superpixel image preprocessing brings considerable speedup to particular object recognition algorithms. The purpose of superpixel algorithms is to cluster pixels based on color and location information. These homogeneous clusters are predefined to be similar in size. Color and location coherence in most cases indicate the association with one particular object. Such an information can be very useful in object recognition algorithms. Recognition process per superpixel can be much more effective compared to per pixel (depending on the superpixel size). We present and evaluate several experiments on Simple Linear Iterative Clustering (SLIC) algorithm, which is the fastest and best quality (boundary adherence) superpixel algorithm. Based on the experiments, we provide our own superpixel algorithm based on the approximation of the Dijkstra search (the shortest path problem) on gradient based images with restricted local searches. Experiments proved lower computational demands, but unfortunately with a decrease in boundary adherence metric.

1 Introduction

In recent years there has been a tremendous increase in camera's resolutions. Particular object recognition algorithms scan images per pixels to evaluate the pixel's assignment to specific object or class. The more pixels in this case naturally mean increase in computational demands.

Pixels are artificially created image representation. Their amount and value depend mostly on the imaging device. The idea of superpixels is to group pixels into bigger clusters to represent image regions. Color and spatial coherence provides high probability that pixels of given superpixel belong to only one particular object. This assumption is used by specific object recognition algorithms which use superpixels instead of pixels [6, 8]. It is still a problem to estimate the proper number of superpixels. This granularity depends on data characteristics. Usually smaller superpixels are chosen to over segment the image to increase the probability, that all pixels from one superpixel belong to only one particular object.

* Doctoral study programme in field: Applied Informatics
Supervisor: Dr. Vanda Benešová, Institute of Applied Informatics, Faculty of Informatics and Information Technologies STU in Bratislava

2 State Of The Art

There are two main types of superpixel algorithms; graph based [4, 10] and gradient [9, 14] based. Graph based algorithms demand more time computation in general, because every pixel is considered as individual graph vertex and the minimum cut algorithm is performed. The first superpixel algorithm based on N-cuts [13] was introduced in [11]. Currently, *Simple Linear Iterative Clustering* (SLIC) is considered to be the State-of-The-Art method based on computational speed and boundary adherence comparison in [1].

The use of superpixels in image segmentation algorithms has several positive aspects, as an enhancement in precision and lower complexity caused by smaller number of image parts to deal with [12]. A method based on histogram of local features computed per superpixel and refined by Conditional Random Field (superpixel neighborhood) outperforms comparable methods on PASCAL VOC 2007 [5]. Another use of superpixels can be found in [15].

2.1 Simple Linear Iterative Clustering

SLIC is a local pixel clustering in 5 dimensional space; 3 dimensions are the image color channels and the other 2 are the x, y coordinates of given pixel [1]. Listing 1 presents the overview of the entire algorithm. First step is to set up the initial superpixel centers along a grid, based on a parameter K, which represents the desired number of approximately equal-sized superpixels. The processed image consists of N pixels, which means there will be approximately N/K pixels in every superpixel region. Grid interval is calculated as $S = \sqrt{N/K}$. To avoid their location on strong edges, they are relocated based on the lowest gradient in 3×3 neighborhood of each center.

For each superpixel center, $2S \times 2S$ neighborhood region is searched and for each pixel distance is measured to the center. Overlapping in this case is the grid size S with neighbor superpixel centers. If the distance is lower than the actual distance of the given pixel to its superpixel, the distance is actualized and the pixel is assigned to this superpixel (during the initialization process, distance for each pixel is set to infinity).

New superpixel centers are recomputed based on assigned pixels as their mean. This process iterates until convergence, the authors claim that 10 iterations are sufficient for almost any image. At the end some isolated pixels from their assigned superpixels occur despite the spatial proximity measure. The distance of these pixels is lower in comparison to other superpixel, mostly due to high color similarity and overlapping regions. Enforce connectivity algorithm (based on Flood fill[1]) eliminates these artifacts. Small ($1/4$ of the declared size) superpixels are merged with neighbor ones, large ones remain.

Listing 1. Simple Linear Iterative Clustering algorithm [1].

```
Set superpixel centers C_k = [l_k, a_k, b_k, x_k, y_k]^T by regular grid with step S.
Move centers in~a 3 × 3 neighborhood to~the lowest gradient position.
REPEAT
  FOR each center C_k DO
    Assign pixels from 2S × 2S neighborhood based on~distance measure
        to~the centers.
  END FOR
  Recompute superpixel centers.
  Compute residual error E.
UNTIL E < threshold
Enforce connectivity.
```

[1] http://en.wikipedia.org/wiki/Flood_fill

2.1.1 Distance Measure

CIELab color model is used by default because of his uniform distance measure between colors. It is inappropriate to use Euclidean distance in this 5D space. If spatial color distances exceed perceptual color distance limit, then they begin to outweigh pixel color similarities which would result in superpixels that do not respect region boundaries. In other words, it is necessary to balance the weights of color similarity and spatial proximity. Equation 1 shows Euclidean distance measure with the weight on proximity distance, where $m = 10$ offers a good balance results.

$$\begin{aligned} d_{lab} &= \sqrt{(l_k - l_i)^2 + (a_k - a_i)^2 + (b_k - b_i)^2} \\ d_{xy} &= \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2} \\ D_s &= d_{lab} + \frac{m}{S} d_{xy} \end{aligned} \tag{1}$$

3 Proposed Modifications

We implement SLIC using the OpenCV library, due to which we speed up mostly the conversion to CIELab color space. We achieve another speed up by using the technique of dynamic programming to calculate some partial results of distance metric calculation. In all tests we use our faster implementation while preserving the quality with parameter $m = 10$ and 10 iterations.

Authors of SLIC tried to use *geodesic* and *adaptively normalized* distance measures with claim of no better performance in speed or boundary adherence. We tried to use two other distances; particularly L1 and Canberra [7], which is similar to Manhattan distance, but with a better predisposition to be used in these kind of problem. Equation 2 represents the Canberra distance metric for vectors p, q. L1 distance performs slightly worse than L2 (in average about 1 – 2%). Canberra distance is slightly better compared to L2 in average about 1%, but it takes 40% more time to calculate. L1 takes up to 4% time more to calculate because the absolute value seems to be more time consuming than the multiplication in L2 distance measure.

$$d_{Canberra}(p, q) = \sum_{i=1}^{n} \frac{|p_i - q_i|}{|p_i| + |q_i|} \tag{2}$$

Precision is measured as boundary adherence on Berkeley BSDS500 database [2]. It represents the percentage of matches between superpixel boundaries and the ground truth contour annotations. Due to lower precision in ground truth annotations (5 different annotations from different people) we dilate the superpixel boundaries twice with kernel 3×3, which means 6 pixels thick contours. Speed was measured for every image (500 images with 481×321 resolution) and averaged. Computation was performed on Intel core i5 3.2GHz processor using a single thread with 10 iterations process.

3.1 Geodesic Distance Experiments

Geodesic distance [1] from pixel $I(p_i)$ to $I(p_j)$ is defined as $\varsigma(I(p_i), I(p_j)) = min_{P \in \Gamma} d(P)$, where Γ is the set of all paths between $I(p_i)$ and $I(p_j)$. The cost associated to path P is given by $d(P) = \sum ||I(p_i) - I(p_{i-1})||_{i=2}^{n}$, where $||I(p_i) - I(p_{i-1})||$ is the Euclidean distance between the CIELab color vectors of pixels p_i and p_{i-1}. The authors claim that connectivity is guaranteed in the x, y plane, which is true in general because the superpixel region is growing from its center continuously. But there are situations mostly when neighbor superpixel "takes" only particularly colored pixels. Distance due to color similarity is low and pixels are reassigned to the other superpixel. The problem is that some pixels remain assigned to the original superpixel and the connectivity is lost as presented in Figure 1. Due to this, it remains necessary to enforce the connectivity. This fact was confirmed by experiments.

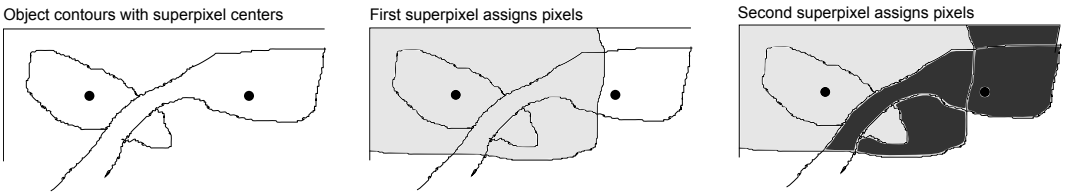


Figure 1. Connectivity need to be enforced while using geodesic distance too. Neighbor superpixel can reassign pixels which can break up the superpixel connectivity.

Table 1. Computation time requirements in seconds for SLIC, geodesic distance using Dijkstra and our approximation to create superpixel of size 100*,* 1050 *and* 11500 *pixels over the image with the resolution of* 481×321 *pixels.*

Method and distance	Superpixel size of 11500 pixels	1050 pixels	100 pixels
SLIC L2	0, 0417864 sec	0, 0571234 sec	0, 0639242 sec
Geo Dijkstra L2	0, 643764 sec	0, 241512 sec	0, 106805 sec
Geo Approximation L2	0, 0373666 sec	0, 0402103 sec	0, 0532248 sec

Based on the geodesic distance definition, we created a graph of pixel differences, where each vertex has 4 neighbors using different distance measures. We design the algorithm as a local graph based search for the shortest path using Dijkstra algorithm [3]. Problem of graph based algorithms, even in local search, is the computational demand (comparison in Table 1). We design an approximation for the shortest path problem for a graph, where every vertex is connected with 4 neighbors (problem of pixels in the image). The approximation first searches in a cross pixels with the superpixel center in its origin (Figure 2) within the S distance in each direction (to the next superpixel center) as in SLIC. The pixels from the cross are assigned to the given superpixel if the distance is lower than the actual distance of the pixel to its superpixel center. Afterwards each quarter is searched individually.

Distance for each pixel is calculated from two x, y directions and shorter path is compared to the actual distance. If the actual distance of the pixel is lower, the search is stopped at the given row/col of that direction. The algorithm is visualized in Figure 2 for better understanding. This approximation takes less time to compute even compared to SLIC, because the search within the region is stopped when the lower actual distance is found. Such an optimization reduces the search region size significantly and it results in lower computational demand. Table 1 presents the logarithmic reduction in time (in seconds) while using Dijkstra minimum path search. SLIC and the Dijkstra approximation slightly increase time demands with the smaller superpixel size, because the number of superpixel rises over the image.

An evaluation was done under the same conditions as mentioned earlier. We also compare different methods used to compute pixel differences; L1 ($\sum |d_i|$), L2 ($\sqrt{\sum d_i^2}$), squared L2 ($\sum d_i^2$),

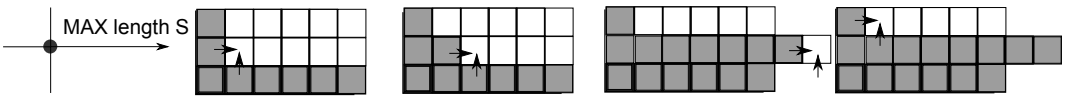


Figure 2. Approximation of the shortest path. Pixels in cross are searched to be assigned to the superpixel while the distance is lower than their actual superpixel. Subsequently search in quarter regions is performed (x or y direction) to assign pixels with a lower distance to current superpixel with the center in the cross origin. It continues until the calculated distance is higher than the actual pixel distance to its superpixel.

Table 2. Comparison in precision: SLIC versus other geodesic distance measures using Dijkstra and CIELab color space to create superpixel of size 100*,* 1050 *and* 11500 *pixels over the image with resolution of* 481×321 *pixels.*

Method	100 pixels	1050 pixels	Superpixel size of 11500 pixels
SLIC L2	93, 8538%	65, 2963%	30, 1762%
Geo Dijkstra L1	92, 4765 %	47, 7081 %	9, 82197%
Geo Dijkstra L2	92, 4561 %	48, 1147%	10, 0904%
Geo Dijkstra Canberra	92, 4476 %	47, 2952%	9, 60821%
Geo Dijkstra Sobel	91, 8%	48, 8012%	10, 3668%
Geo Dijkstra L2 squared	93, 6549%	57, 6051%	15, 9092%
Geo Dijkstra $\sqrt{\sum d_i^3}$	93, 3499%	54, 6058%	13, 4821%
Geo Dijkstra $\sqrt{\sum d_i^4 or d_i^3}$	93, 4277%	56, 219%	14, 8306%
Geo Dijkstra $\sum d_i^3$	93, 5791%	60, 0273%	18, 7435%
Geo Dijkstra $\sum d_i^4 or d_i^3$	93, 353%	60, 2965%	19, 3017%
Geo Dijkstra Sobel squared	93, 0698%	57, 9863%	16, 3495%

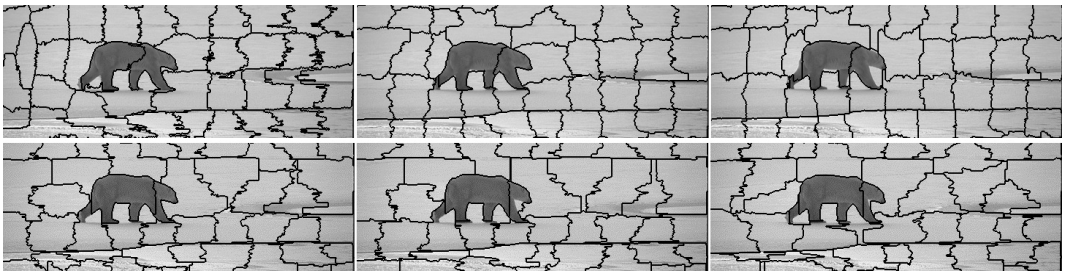


Figure 3. Visual comparison of created superpixels (left to right: SLIC, Dijkstra, Dijkstra approximation using CIELab color space). Geodesic distance has more compact superpixels with not so fuzzy boundaries until we use the adjusted distance calculations (second row: $\sum x_i^3$*,* $\sum (x_i^4 | x_i^3)$*, Sobel squared).*

Canberra and Sobel kernel 3×3. Sobel was taken into consideration, because its gradients have also the ability to describe the pixel difference (x, y directions are always calculated separately). Due to experiments with squared L2; we analyzed that if we increase the data difference, we can achieve higher boundary adherence results. We tried to modify the L2 and L2 squared distance calculation by increasing the number under the square root using the power of 3 and 4 instead of 2. Such a modification meant an increase in boundary precision up to 10%. We also tried to do power of 2 of Sobel values, which also provides better quality results than the original values with up to 8% increase. We achieved the best quality by using a combine distance method. If the calculated original difference is lower than 100, we do power of 4 and the power of 3 otherwise to increase the smaller values and magnify the edges. Difference in precision while using these different distance measures is presented in Table 3.1. Our experiments confirm that CIELab color space provides the best performance with difference compared to RGB up to 2% of precision. Our Dijkstra approximation provides similar results as the original Dijkstra algorithm, difference in quality is up to 2% with significant time spared.

Figure 3 shows the difference in boundary properties. SLIC and adjusted distance do not have such coherent regions as L2 distance measure while using Dijkstra and its approximation. Experiments on distance measures do not provide proper coherent boundaries either, however with better boundary adherence as in SLIC. Figure 4 presents the comparison of all provided tests on SLIC, Dijkstra and approximation using color spaces: Gray scale, RGB and CIELab. The best precision

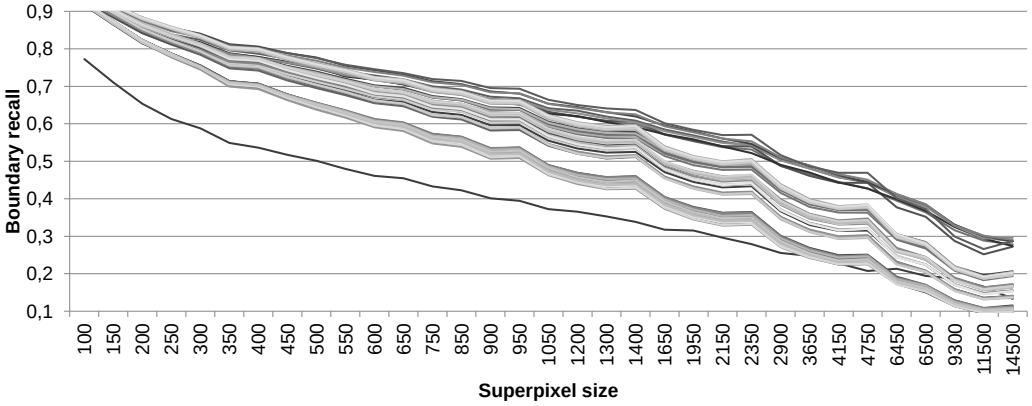


Figure 4. Comparison in boundary adherence of SLIC, Dijkstra and the proposed approximation while using different color models (Gray scale, RGB, LAB) and different distances.

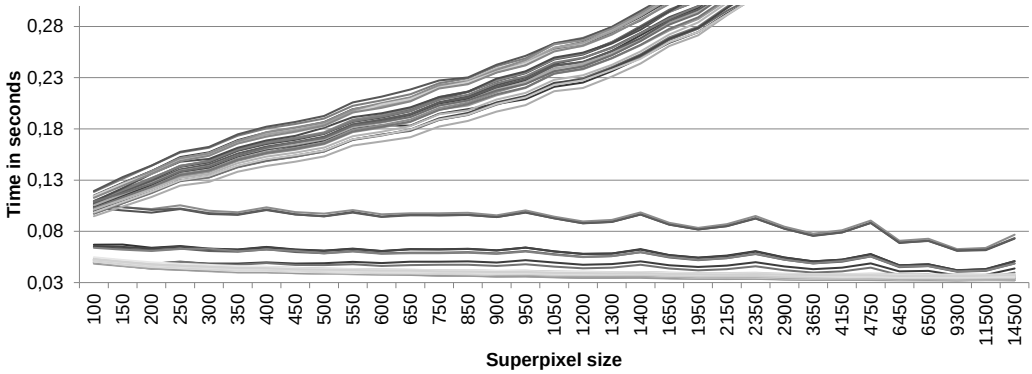


Figure 5. Comparison in time computation of SLIC, Dijkstra and the proposed approximation while using different color models (Gray scale, RGB, LAB) and different distances.

represents the upper line group of SLIC which have up to 20% better performance compared to worst Dijkstra and 10% compared to the best Dijkstra results achieved by combining a distance measure in CIELab color space. The groups of lines below, represent Dijkstras and theirs approximations gathered based on different distance measures. The worst results are with SLIC on Gray scale using L1 distance measure. Figure 5 represents the time dependecies of evaluated algorithms, as presented earlier geodesic Dijkstra search has an exponential dependence on the superpixel size (which also means a smaller number of superpixels). Geodesic distance measure using Dijkstra approximation is almost in all cases faster than SLIC (represented by a group of lines up to them).

4 Conclusion

We did experiments on geodesic distance measure to provide reliable superpixel image segmentation algorithm. We tried several distance measures to calculate the pixel differences as L1, L2, Canberra, Sobel, L2 squared, Sobel squared and adjustments to L2 and L2 squared distances as $\sqrt{\sum d_i^3}$, $\sqrt{\sum (d_i^4|d_i^3)}$, $\sum d_i^3$, $\sum (d_i^4|d_i^3)$ while using different color spaces; Gray scale, RGB and CIELab and two pixel assignment algorithms Dijkstra and our Dijkstra approximation. We conclude; CIELab color space achieves better performance up to 2% compared to RGB. Also our approximation provides

similar results to the original Dijkstra with the difference up to 2% and mostly that the best quality is provided by the combined distance measure $\sum (d_i^4 | d_i^3)$, where the pixel difference is emphasized by the power of 4 if it is lower than 100 and by the power of 3 otherwise, but the quality compared to SLIC is still lower to 10%. The computation time is better up to 20% due to smaller search region size, which is adjusted based on the actual pixel assignments and distances.

Acknowledgement: This work was partially supported by the Cultural and Educational Grant Agency of Slovak Republic, grant No. KEGA 068UK-4/2011.

References

[1] Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Suandsstrunk, S.: SLIC Superpixels Compared to State-of-the-Art Superpixel Methods. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2012, vol. 34, no. 11, pp. 2274–2282.

[2] Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour Detection and Hierarchical Image Segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2011, vol. 33, no. 5, pp. 898–916.

[3] Dijkstra, E.: A note on two problems in connexion with graphs. *Numerische Mathematik*, 1959, vol. 1, pp. 269–271.

[4] Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient Graph-Based Image Segmentation. *Int. J. Comput. Vision*, 2004, vol. 59, no. 2, pp. 167–181.

[5] Fulkerson, B., Vedaldi, A., Soatto, S.: Class segmentation and object localization with superpixel neighborhoods. In: *Computer Vision, 2009 IEEE 12th International Conference on*, 2009, pp. 670–677.

[6] He, X., Zemel, R.S., Ray, D.: Learning and incorporating top-down cues in image segmentation. In: *Proceedings of the 9th European conference on Computer Vision - Volume Part I*. ECCV'06, Berlin, Heidelberg, Springer-Verlag, 2006, pp. 338–351.

[7] Lance, G.N., Williams, W.T.: Computer Programs for Hierarchical Polythetic Classification ("Similarity Analyses"). *The Computer Journal*, 1966, vol. 9, no. 1, pp. 60–64.

[8] Levinshtein, A., Dickinson, S., Sminchisescu, C.: Multiscale symmetric part detection and grouping. In: *Computer Vision, 2009 IEEE 12th International Conference on*, 2009, pp. 2162–2169.

[9] Levinshtein, A., Stere, A., Kutulakos, K.N., Fleet, D.J., Dickinson, S.J., Siddiqi, K.: TurboPixels: Fast Superpixels Using Geometric Flows. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2009, vol. 31, no. 12, pp. 2290–2297.

[10] Liu, M.Y., Tuzel, O., Ramalingam, S., Chellappa, R.: Entropy rate superpixel segmentation. In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, 2011, pp. 2097–2104.

[11] Ren, X., Malik, J.: Learning a classification model for segmentation. In: *Proc. 9th Int'l. Conf. Computer Vision*. Volume 1., 2003, pp. 10–17.

[12] Rohkohl, C., Engel, K.: Efficient Image Segmentation Using Pairwise Pixel Similarities. In Hamprecht, F., Schnörr, C., Jähne, B., eds.: *Pattern Recognition*. Volume 4713 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 2007, pp. 254–263.

[13] Shi, J., Malik, J.: Normalized Cuts and Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2000, vol. 22, no. 8, pp. 888–905.

[14] Vedaldi, A., Soatto, S.: Quick Shift and Kernel Methods for Mode Seeking. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, 2008.

[15] Vezhnevets, A., Ferrari, V., Buhmann, J.: Weakly supervised semantic segmentation with a multi-image model. In: *Computer Vision (ICCV), 2011 IEEE International Conference on*, 2011, pp. 643–650.

Improving Binary Feature Descriptors Using Spatial Structure

Michal KOTTMAN*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
kottman@fiit.stuba.sk

Abstract. Feature descriptors are used to describe salient image keypoints in a way that allows easy matching between different features. Modern binary descriptors use bit vectors to store this information. These descriptors use simple comparisons between different keypoint parts to construct the bit vector. What they differ in is the arrangement of keypoint parts, ranging from random selection in BRIEF descriptor to human vision-like pattern in the FREAK descriptor. A recent descriptor D-Nets shows that line-based arrangement improves recognition rate for feature matching. We show that by extending the comparisons to line arrangement better describes the spatial structure surrounding the keypoint and performs better in standard feature description benchmarks.

Amended version published in Information Sciences and Technologies Bulletin of the ACM Slovakia, Vol. 5, No. 2 (2013), pp. 27-31.

* Doctoral degree study programme in field: Applied Informatics
Supervisor: Assoc. Professor Martin SŠperka, Institute of Applied Informatics, Faculty of Informatics and Information Technologies STU in Bratislava

Software Engineering

Use of Design Patterns in Modeling Service Oriented Architecture

Adrián FEJEŠ*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
fejoo89@gmail.com

Abstract. In SOA, like in other areas of software engineering, design patterns were identified. Nowadays, the modeling of design patterns is mostly realized by standard UML diagrams. However, UML is a general modeling language and therefore not suitable for representing domain specific information. In this paper we present a metamodel and UML profile for representing SOA design patterns in models. This metamodel can be useful for representing design patterns in models in a way, that allows their identification and utilization (e.g. model validation, code generation). We demonstrate the use of metamodel by creating models of a simple banking system and by generating WSDL document from a part of this model.

1 Introduction

Service-Oriented Architecture is an architectural style for building systems based on interacting services. In SOA, like in other areas of software engineering, using design patterns has become very popular. Design patterns provide general repeatable solutions to commonly occurring problems and can be effectively used for the modeling of software systems.

Service-Oriented Architecture systems tend to be complex and the use of design patterns could increase the quality of the development process and the overall software system. Nowadays, many systems are mostly modeled by standard UML diagrams. However, it is difficult to recognize that these diagrams are correct and that all good design practices are followed. In our opinion, some of these problems could be solved with utilization of design pattern. An overview of SOA design patterns is published in [2]. This publication describes a lot of designs patterns which provide best practices in development of SOA systems. These patterns are described in an informal way which can lead to ambiguity and inaccuracy. We need pattern representation which supports (semi-)automatic utilization of design pattern in modeling of SOA based systems. Such representation could be used to avoid ambiguity and to utilize design patterns (e.g. model validation, code generation).

Another problem of modeling with UML is, that UML is a general modeling language and therefore not suitable for representing domain specific information. Modeling of SOA design

* Master degree study programme in field: Software Engineering
Supervisor: Roman Šelmeci, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

patterns, however, needs the capability to capture information specific to service-orientation paradigm (e.g. service provider, service consumer, service contract).

In this paper we propose a metamodel and UML profile for representing SOA design patterns in models. This metamodel can be useful for representing design patterns in models in a way, that allows their identification and utilization. One of our main goals is to check if models based on the proposed profile comply with the constraints of this profile. If a model is valid against the profile constraints, we can assume that the model does not violate principles of design patterns defined by these constraints. We focus on service contract design patterns, but the metamodel can be easily extended to support other related design patterns.

The paper is structured as following. Section 2 describes related work in this area. In section 3 we describe the proposed metamodel and UML profile for representing SOA design patterns in models. Section 4 contains the use of proposed approach by creating models of a simple banking system in which design patterns are present, and by generating WSDL document from a part of this model. In section 5 the evaluation of the proposed solution is given. Finally, Section 6 presents our conclusions and plans for future work.

2 Related work

This section provides an overview of works related to patterns used in the field of software modeling. There have been done some research on formalization of design patterns and on use of design patterns in modeling software systems.

Authors in [4] present the *Design Pattern Modeling Language (DPML)*, a language supporting the specification of design pattern solutions and their instantiation into UML design models. It is important to notice that DPML can only be used to model the generalized solutions proposed by design patterns, not complete design patterns. The major benefit of DPML is the ability to work with design patterns in conjunction with UML.

Work [3] proposes modeling architecture for patterns using UML profiles. According to authors it is not possible to define a semantic for all patterns in a single profile. They believe that it is necessary to define a profile for each pattern, where in each profile the semantic of a particular pattern is described. One benefit of this solution is that it is based on UML Profile, so the solution is fully compatible with the UML standard. Another advantage is the proposed hierarchy between levels of profiles allowing the reuse of definitions.

Authors of [1] present the use of *Service Oriented Architecture Modeling Language* for specifying services. SoaML is a UML profile and a metamodel for the design of services within a service-oriented architecture. The main goals of SoaML are to support the activities of service modeling and design and to fit into an overall model-driven development approach. Authors discuss 3 different approaches to specifying services: simple interface based approach, service contract based approach and service interface based approach. They have presented a set of practical modeling guidelines for how to align the different approaches to specifying services using SoaML.

The SoaML language presented in previous paragraph can be appropriate for modeling SOA design patterns. An attempt to use SoaML to model SOA design patterns is made in [5]. In this work authors propose a formal architecture-centric approach that aims to model message-oriented SOA design patterns with the SoaML language. According to authors the main reasons for using SoaML is, that it is a standard language defined by OMG and diagrams used in this language allow representing structural features as well as behavioral features of SOA design patterns. The proposed approach is illustrated through the modeling of *Asynchronous Queuing* pattern.

Most of the works in the area of modeling design patterns concentrate on object-oriented design patterns and the support for SOA design patterns is poor. Therefore, in this work we propose a UML profile and metamodel for modeling SOA design patterns.

3 UML profile and metamodel for representing SOA design patterns in models

A metamodel is a special kind of model that specifies the abstract syntax of a modeling language. It specifies the rules for creating models by describing meta-elements, their attributes and relationships. The UML provides a standard mechanism for general extending of UML metamodel by creating UML profiles. UML Profile enables the extending and adapting of UML to a platform or domain by using a package stereotyped as *"Profile"*. This package provides three mechanisms for extending UML: stereotypes, tag values and constrains. We created UML profile in several steps.

Firstly, the main elements of the metamodel were identified from the description and analysis of design patterns. During the analysis we identified the main attributes of the elements as well. We believe that the specification of values of some identified attributes can be automatic. For example, if the *Service* element is related to at least one *Capability* element, whose *is Redundant* attribute is set to true, the *hasRedundantCapabilities* attribute of *Service* element can be automatically set to true as well. The proposed metamodel is shown in the Figure 1.

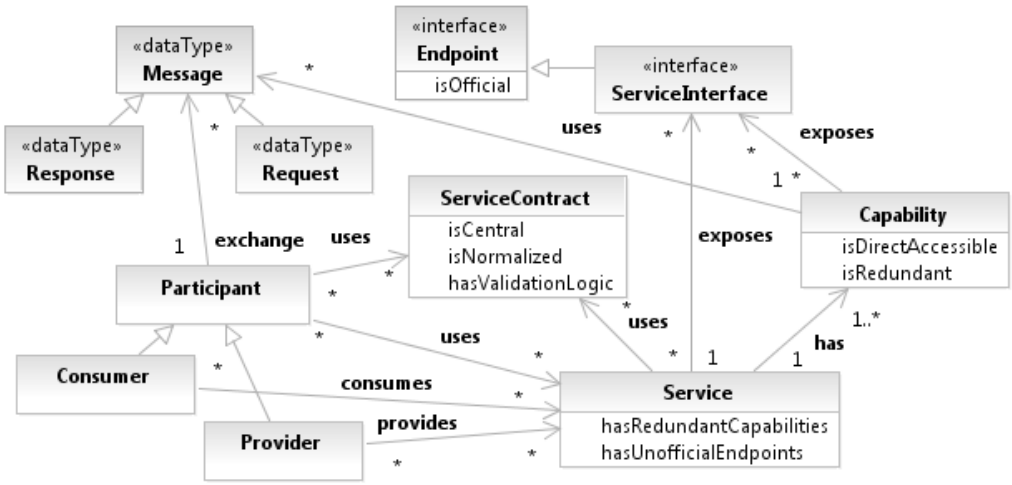


Figure 1. Metamodel for representing SOA design patterns in models.

Then we created a profile by defining stereotypes based on the proposed metamodel. Stereotypes were defined for each element of metamodel and their relationships. In order to maintain clear relationship between the metamodel and profile, the name of the stereotype is the same as the name of the metaelement which was extended.

The last step to define the profile is the specification of constraints. These constraints reflect rules defined in the description of design patterns and support the creation of design patterns with correct structure in models. Models based on this profile may serve as a foundation for automated code generation. Therefore, they require a precise and unambiguous meaning. For the specification of constraints we used the OCL language[1]. We focused on the definition of the associations' cardinalities and on the definition of type and initial value of attributes. The origin (design patterns) is given for each constraint. In the Figure 2 a constraint of the stereotype *Service* is illustrated.

[1] http://www.omg.org/spec/OCL/2.2/PDF

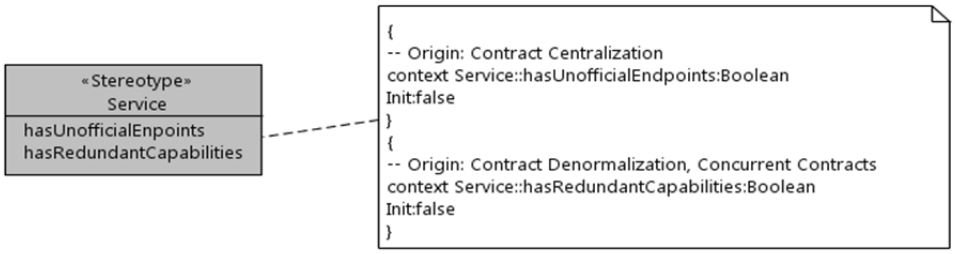


Figure 2. OCL constraints for the stereotype Service.

For the stereotype *Service* two constraints are defined, both specifying the initial values of attributes (in this case the value *false*). The constraint of the attribute *hasUnofficialEnpoints* was identified from the description of *Contract Centralization* design pattern and the constraint of the *hasRedundantCapabilities*attribute was identified from the description of *Contract Denormalization* and *Concurrent Contracts* design patterns.

After the profile was created we realized that our profile has many elements in common with the SoaML language. This means that during the analysis of design patterns we identified elements that conform to concepts used by the SOA community.

4 A Case Study

The use of proposed profile we illustrate by creating models of a simplified example of banking system. The sample banking system is made up of several subsystems, which are integrated by services. The main purpose of the system is to provide and consume information about customers and their accounts. The model of the system (Figure 3) contains one design pattern: *Denormalized Contract*. In the Figure 3 the standard UML model of the banking system is illustrated.

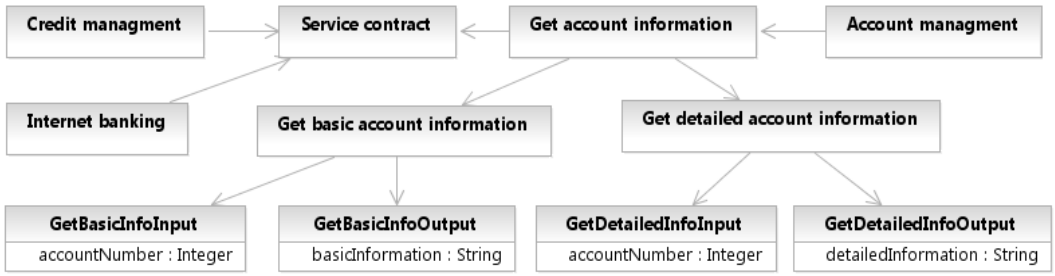


Figure 3. Standard UML model of the banking system.

The model in the Figure 3 contains only standard UML elements. It means that the model does not capture any domain-specific information (in our case information specific to service-orientation). Without additional information it is difficult to recognize who is the consumer or provider of the service (resp. what the elements of model represent). Because of the absence of additional information, computer (or human) cannot recognize that model contains design patterns and their utilization is difficult or even worst impossible. Figure 4 illustrates the UML model of the banking system; in this case the proposed profile was applied.

The model in the Figure 4 captures domain-specific information by applying the proposed profile. It is clear, that what a role each element in the model plays, resp. what relationships between the elements exist. This additional information about roles and relationships may make it

possible to recognize design patterns. For example, the presence of several consumers and capabilities, and their relationships to other elements of model in the Figure 4,may indicate that the model probably contains the *Denormalized Contract* design pattern. Applying the profile to the model may lead to a more efficient and precise representation of the system which could reduce the time needed to understand it and could allow utilization of design patterns (e.g. model validation, source code generation).

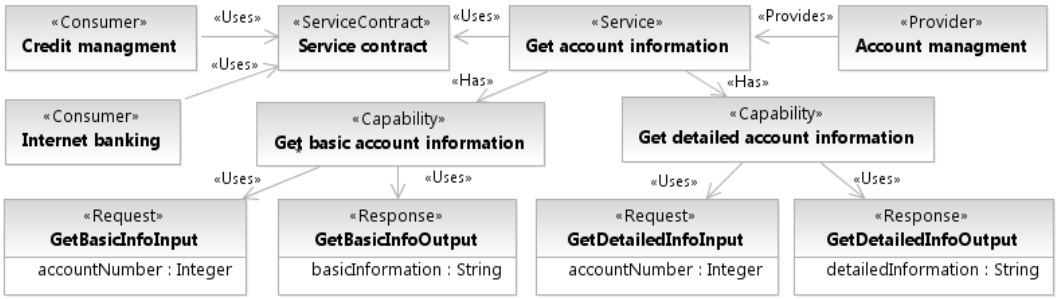


Figure 4. Model of the banking system based on proposed profile.

5 Evaluation

We had two main goals during the evaluation phase. First goal was to find out if our approach is suitable for validating models with SOA design patterns. Second goal was to find out if our approach can form a base for source code generation or generation of other types of document during development of SOA based systems.

In the previous section we showed, that the proposed UML profile is appropriate to capture domain-specific information (in our case information specific to service-orientation). This additional information can increase the success and speed of the process of design pattern recognition. An issue related to the proposed approach is to check if models based on a profile comply with this profile and if they really contain design patterns. If we are able to recognize the design patterns, we can verify if a solution is based on verified good practices. On the other hand, if we cannot identify the patterns there are some options: model does not contain pattern, model contains other variation of patterns for which OCL constraints are not already defined, or there are places where the model violates defined constraints of profile and that way we can identify software design bugs. As shown in the Section 3, the proposed UML profile is supplemented by constraints specifications. These constraints are needed, because they formally define the constraints of the proposed stereotypes. To validate these constraints it is important to find an appropriate tool, we decided to use the Eclipse MDT OCL[2].

We made an attempt to validate model in the Figure 4. The validation process was semi-automatic, this means:1) we must manually give the OCL constraints as input to the Eclipse MDT OCL tool and 2) it automatically validated the model against OCL constraints. The result of validation showed, that the model in the Figure 4 complies with the constraint of metamodel and therefore we can assume, that it is built on verified good practices. Then the same validation process was applied to the model that violates the profile's constraints. In this case the validation failed and we were able to identify places where the model does not comply with the profile's constraints.

Adding domain-specific information to models makes it possible to generate source code from the model. As our solution focuses on service contract design patterns we decided to generate

[2] http://www.eclipse.org/modeling/mdt/?project=ocl

WSDL[3] document. We identified mapping between the elements of WSDL document and elements of proposed profile. The generation of WSDL document is realized by XSLT transformation. The resulting document is not complete, for example it does not contain information about the transfer protocol and about the message format. This information can be specified by using other design patterns such as *Canonical Protocol* or *Dual Protocol*. Another option is the use of predefined values (e.g. HTTP for transfer protocol or "document" for message format).

6 Conclusions

In this paper we presented a metamodel and UML profile for representing SOA design patterns in models. We concentrated on service contract design patterns but the profile can be easily extended to support related design patterns (e.g. *Logic Centralization*, *Service Composition*).

We showed, that the proposed approach can be used to model SOA based systems and supports verification of system's models according to design patterns. We compared the standard UML model of a simple banking system and the model created by applying our profile. The model with the profile contained domain-specific information, which can be useful to facilitate design patterns. For the validation of models we used the Eclipse MDT OCL tool, which supports OCL validation. We also showed that the proposed profile can be used to support implementation of SOA based systems by generating WSDL documents from the models.

Validation and automated WSDL generation based on the proposed profile seems to be promising approaches to improve the development process of SOA based systems. In the future, we will work on improving the validation of OCL constraints and WSDL generation. We would like to extend our approach to support other related design patterns.

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic grant No. VG1/1221/12.

References

[1] Elvesæter, B., Berre, A.J., Sadovykh, A.: Specifying Services using the Service Oriented Architecture Modeling Language (SOAML): A baseline for Specification of Cloud-based Services, *SciTePress*, Portugal (2011), pp. 276–285.

[2] Erl, T.: SOA Design Patterns. Prentice Hall, (2009).

[3] Garis, A., Riesco, D., Montejano, G.: *Defining Patterns Using UML Profiles*. N. C. Debnath Winona State University, Department of Computer Science Winona, MN 55987 USA, (2002).

[4] Mapelsden, D., Hosking, J., Grundy, J.: *Design Pattern Modelling and Instantiation using DPML*, Department of Computer Science, University of Auckland, Auckland, New Zealand, (2002).

[5] Tounsi, I., HadjKacem, M., HadjKacem, A., Drira, K.: An Approach for Modeling and Refinement of SOA Design Patterns with Event-B Method, In: *The 9th workshop on Methods for the Adaptive Distributed Software (METHODICA-II'2012)*, Hammamet, Tunisia, (2012).

[3] http://www.w3.org/TR/2002/WD-wsdl12-20020709/

Source Code Authorship Detection Using User Modeling

Maroš MARŠALEK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
maros.mars@gmail.sk

Abstract. In this paper we briefly describe problem area of source code authorship detection with focus on coding style and examine current state of research in this field. We present our own solution that combines both static characteristics of coding style extracted from source code as well as dynamic characteristics captured during code development in an Integrated Development Environment. We describe a composite model that we designed to store these characteristics as well as a unique method for coding style comparison based on k-nearest neighbor algorithm. Results from initial experiments are also presented.

1 Introduction

Identification of source code author has become a common task in these days. It is often dealt with in the field of research and education in order to detect plagiarism. However, plagiarism detection is not the only case where the rightful author needs to be detected. Authorship disputes, proof of authorship in court or even tracing the source of code left in the system after a cyber attack also utilize authorship detection methods.

Some of these methods are based on comparison of source code authors' style. The general idea behind this approach is that every author keeps a certain style of source code writing. This style is different for different authors and is preserved in all source codes written by the same author.

Even though source code is a technical text often written by a set of guidelines, it provides enough room to capture the style of its author. There are multiple characteristics that contain information about the style e.g. formatting, commentaries, naming conventions, mistakes etc. Figure 1 illustrates differences in styles of two source code authors. It contains two implementations of the factorial function created by two different authors. The differences are apparent and it is possible to utilize these differences to identify the most likely author.

In order to utilize author's style for identification, we need to capture his or her style in a model. This model receives characteristics of author's style extracted from source code, which

* Master degree study programme in field: Software Engineering
Supervisor: Assoc. Professor Daniela Chudá, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 239–244.

we are certain was written by the author. After the model is filled with adequate amount of characteristics, we can compare characteristics extracted from analyzed source code with characteristics in the model and calculate the likelihood of authorship.

```
// Factorial takes an integer as an input and returns
// the factorial of the input.
// This routine does not deal with negative values!

int Factorial (int Input)
{
       int Counter;
       int Fact;
       Fact=1;  // Initalises Fact to 1 since factorial 0 is 1
       for (Counter=Input; Counter>1; Counter=Counter-1)
       {
                    Fact=Fact*Counter;
       }
       return Fact;
}
```

```
int f(int x){
int a, y=1;
if (!x) return 1; else return   x*f(x-1);}
```

Figure 1. Illustration of different coding styles.

2 Related work

First attempts to identify the author or at least gain some information about author's intentions were performed in 1989 during the analysis of a computer virus [3, 4]. This virus infected great amount of computers running BSD operating system and was quite successful. Author of this virus was not identified but the analysis of style used in the source codes discovered some interesting facts. The mistakes, used data structures and other unusual constructs disproved that the author was an expert programmer as everyone thought.

After these discoveries, more research papers appeared with intention to discover and list characteristics that would provide information about author's style. Basic list of characteristics was published in [5] and updated in [2].[1] This list contained groups of characteristics such as formatting, variable names, commenting style, errors, code metrics etc.

One of the first methods that used author's style for authorship detection was published in [3]. This method used a predefined set of characteristics to model author's style. The predefined set contained characteristics such as Percentage of open curly brackets ({) that are alone in a line, Mean local variable name length or Percentage of "void" function definitions. An experiment was performed to calculate the success rate of this method and the resulting rate was at 73%. All source codes for this experiment were written in C language by 29 different authors. Results of this research paper were considered a success and showed the potential of authorship detection based on coding style.

Multiple papers with similar methods were published in the following years. One of the most successful was proposed in [1] and was based on extraction of most used byte level n-grams. This method differs from the others as it does not use a predefined set of characteristics and relies on a low-level approach. This allows it to theoretically capture all characteristics of coding style in comparison to a utilization of a predefined set of characteristics, where not all possible coding style characteristics can be covered. The experiments performed by the authors showed greater performance over methods based on predefined set of characteristics and they also proved that this method is language independent.

2.1 Evaluation

All presented methods used finished source codes as a base for coding style characteristics. Such characteristics can be referred to as static. Despite static characteristics being wildly used in the research of source code authorship detection, they bring crucial disadvantages that limit accuracy. The first is characteristics disguise caused by coding standards utilization or utilization of a common code formatter. The other one is team development where it is typical for multiple authors to develop shared source codes.

We believe that these disadvantages can be alleviated by providing additional characteristics of coding style that can be captured during the work on source codes in an IDE (*Integrated Development Environment*). We call these characteristics dynamic since they capture user's style of interacting with an IDE. Our goal is to design a method that will outperform currently available methods by adding additional characteristics to user's model of coding style.

3 Our solution

In this chapter we describe our solution for source code authorship detection utilizing coding style (see Figure 2). It is divided into three main parts, which are focused on characteristics that we capture, user model for coding style and authorship detection using the designed model.

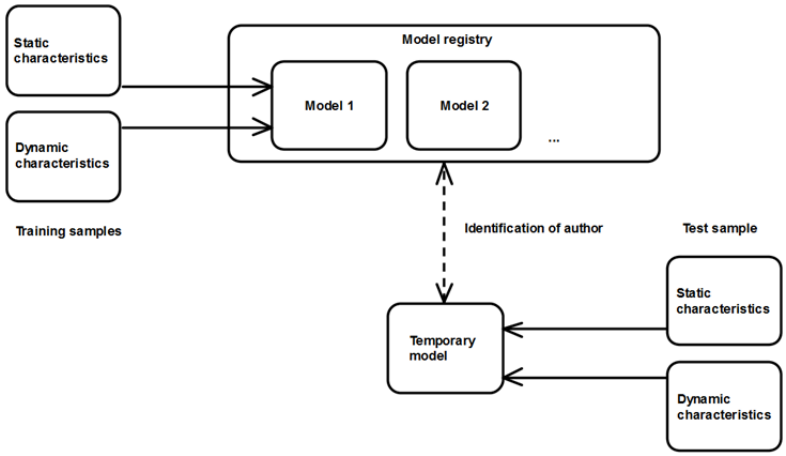


Figure 2. Overview of proposed solution.

3.1 Characteristics

One of the key aspects of user modeling is to choose a set of characteristics that will be captured in the model. In our case it was set of static and dynamic characteristics to capture coding style.

With static characteristics we took the approach proposed in [1] as it was used by one of the most successful methods for source code authorship detection. Static characteristics are therefore represented by X most used byte level N-grams. We decided to take this approach for two main reasons and they were language independency and ability to capture all static characteristics without having a predefined set of characteristics.

In case of dynamic characteristics we could not utilize any existing approach since there are no papers focused directly on user modeling of coding style in IDE. From the analysis of papers dealing directly or indirectly with IDE usage, we were able to define a set of dynamic characteristics divided into 7 groups. We believe that these groups of characteristics together capture dynamic coding style but we need to perform experiments in order to analyze the impact of these individual groups on authorship detection. The groups of dynamic characteristics are:

- Orientation in code editor (e.g. ratio of keyboard and mouse orientation or frequency of keyboard shortcuts usage for advanced orientation events frequency)
- Code editing (e.g. frequency of copy/paste actions or move lines actions)
- File usage (e.g. open file action or save file actions frequency)
- IDE assistance utilization (e.g. average time of content assist dialog open or manual open of content assist frequency)
- IDE command utilization (e.g. refactor or search commands usage frequency)
- Dynamics of work (e.g. frequency of performed actions)
- Undo / Redo usage (undo and redo actions frequency)

3.1.1 Dynamic characteristics capturing

Capturing static characteristics in our solution is straightforward as we need to extract byte level N-grams from the source code. On the other hand, capturing of dynamic characteristics is more difficult as we need to capture author's activity in IDE. We have considered multiple existing loggers for 2 popular IDEs (*Eclipse IDE, MS Visual Studio*) and we have chosen the *Fluorite logger* for *Eclipse IDE* [6]. It produces *xml* based log files with captured activity and outperforms other available activity loggers for IDE mainly in terms of number of different actions captured and installation and usage simplicity.

3.2 User model

We have designed a composite model to store presented characteristics. This model is divided into two sub-models to separate different types of stored characteristics. One of them stores only static characteristics and the other stores only dynamic characteristics.

The static sub-model stores vector of most used and sorted byte level N-grams and its design was adopted from [1] since we took approach proposed in this paper.

For dynamic characteristics, we have designed a custom sub-model that is further divided into 7 dynamic sub-models. These dynamic sub-models conform to the groups of dynamic characteristics that we defined. Each model stores a vector of key-value pairs where the key is a certain characteristic and the value is a numeric evaluation of that characteristic e.g. Average length of a debug session, Frequency of keyboard based orientation in code or Average delay between actions executed in IDE.

The separation of characteristics in different sub-models allows for utilization of different comparison method for characteristics in every sub-model or performing of experiments with different active sub-models and thus determining the influence of different groups of characteristics on authorship detection.

3.3 Authorship detection

After the user models are developed, we can perform the authorship detection on a test set of static and dynamic characteristics. The detection is performed by comparing characteristics from available user models with characteristics from a test set. The author of model containing the most similar characteristics is presented as the most likely author of the test set.

This description conforms to the general idea of k-nearest neighbor algorithm and in our case the proximity in space is calculated as similarity of coding styles captured in user models. If we extract characteristics from the test set, store them in a temporary model and project it into space with all user models, it will be placed somewhere near the model with most similar characteristics. The positions obtained by the models depend on the values stored in the models, which in our case represents the coding style. The challenging part of this approach is designing of a suitable distance function for proximity calculation.

We have designed a multi-level distance function suitable for our model, which combines the results of multiple distance functions designed specifically for the sub-model. On the top level, the function combines results from distance function for static sub-model and from distance function for dynamic sub-model:

$$\text{Distance} = \text{Weight}_\text{s} * \text{Distance}_\text{s} + \text{Weight}_\text{d} * \text{Distance}_\text{d}$$

where *Distance*$_x$ represents result of the static distance function, which calculates similarity by comparing vector of most used byte level N-grams. The original distance function designed in [1] considered only the size of vectors' intersection. We enhanced the function by calculating difference based on position of the same N-grams:

$$\text{Distance}_\text{s} = \sum_1^\text{n} |\text{Position}(\text{ngram}_\text{i})_{\text{in model 1}} - \text{Position}(\text{ngram}_\text{i})_{\text{in model 2}}|$$

The *Distance*$_d$ value represents result of the dynamic distance function. It combines results from distance functions for every dynamic sub-model and is based on same principle as the top level distance function:

$$\text{Distance}_\text{d} = \sum_1^\text{n} \text{Weight}_\text{n} * \text{Distance}_\text{dn}$$

Finally the *Distance*$_{dn}$ results for every dynamic sub-model are calculated as a sum of differences of every value found in that sub-model.

We have also introduced a configurable weighing system into the functions. It allows us to reflect the influence of different coding style groups of characteristics in the process of authorship detection.

4 Experiments

We have implemented a prototype of our solution and performed first experiments. In this chapter we will present the results of the most interesting one, which is identification of a user based only on dynamic characteristics from an IDE.

This experiment was performed on a group of 7 programmers with 5-10 years of experience and the characteristics were captured within 2-3 weeks. The goal of this experiment was to determine if the dynamic characteristics from an IDE can be used for author identification. The results are show in table 1 and for each group of characteristics there is a percentual evaluation of its user identification success rate.

Table 1. Results from user identification with dynamic characteristics.

Characteristics group	**User identification success rate**
Orientation in editor	*71%*
Code editing	*72%*
File usage	*45%*
IDE assistance utilization	*33%*
IDE command utilization	*33%*
Dynamics of work	*20%*
Undo / Redo utilization	*32%*
Compound	*62%*

The results are not optimal and vary from 20% to 72% with compoun success rate (identification using all groups together) at 62%. On the other hand these results look promising since the distance function weight configuration was set to default (all weights set to 1) and the model was not optimized. We think that this experiment proved the ability of dynamic characteristics to

identify its author. We also think that in further experiments with optimized model and optimized distance function configuration we will achieve even better results.

We are planning on performing additional experiments to fully evaluate our solution and these experiments are:

- Authorship detection using only static characteristics in order to optimize the static model and calculate static success rate.
- Repeat authorship detection using only dynamic characteristics in order to optimize the dynamic model and calculate dynamic success rates for every group of characteristics.
- Authorship detection using static and dynamic characteristics in order to calculate overall success rate and compare it to static success rate.
- Authorship detection using static and dynamic characteristics in one project with multiple authors in order to calculate the distribution of work amongst the authors.

5 Conclusions

We have briefly described the field of authorship detection in source code with focus on coding style. Afterwards we have examined current state and trends in this field and based on this knowledge we have proposed a source code authorship method that captures coding style using static and dynamic characteristics. We have designed a model to store coding style and a method for identification that is based on comparison of captured coding styles. We have performed initial experiments and presented the results of one of them. These results confirm our assumptions and suggest that our solution can be further developed and final experiments can be performed. In the future, the model might need to be optimized by updating the set of dynamic characteristics and determining adequate sizes for both N-gram size and list of N-grams in static model. Also correct weights for distance functions need to be found and additional experiments to accomplish these tasks need to be performed.

Acknowledgement: This work was partially supported by the grant No. VG1/0971/11 and it is the partial result of the Research & Development Operational Programme for the project Research of methods for acquisition, analysis and personalized conveying of information and knowledge, ITMS 26240220039, co-funded by the ERDF.

References

[1] Frantzeskou, G. et al.: Effective Identification of Source Code Authors Using Byte-Level Information. In: *Proceedings of the 28th international conference on Software engineering*. New York: ACM, (2006), pp. 893–896.

[2] Gray, A. et al.: Software Forensics: Extending Authorship Analysis Techniques to Computer Programs. In: *Proceedings of the 3rd Biannual Conference of the International Association of Forensic Linguists (IAFL)*. Durham NC,USA, (1997), pp. 1–8.

[3] Krsul, I., Spafford, E.H.: Authorship Analysis: Identifying The Author of a Program. *Computers Security*, (1995), vol. 16, no. 3, pp. 233–257.

[4] Spafford, E.H.: The Internet Worm Program: An Analysis. *SIGCOMM Computer Communication Review*, (1989), vol. 19, no. 1, pp. 17–57.

[5] Spafford, E.H., Weeber, S.A.: Software Forensics: Can We Track Code to its Authors? *Computers and Security*, (1993), vol. 12, no. 6., pp. 585–595.

[6] Yoon, Y., Myers, B.A.: Capturing and Analyzing Low-Level Events from the Code Editor. In: *Proceedings of the 3rd ACM SIGPLAN workshop on Evaluation and usability of programming languages and tools*. New York: ACM, (2011), pp. 25–30.

Using Aspect-Oriented Change Realization to Introduce and Document Changes in Object-Oriented Models

Ľuboš STARÁČEK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
lubostar1@gmail.com

Abstract. Change requests appear during software development and maintenance. To implement change requests that exhibit crosscutting it seems to be appropriate to use aspect-oriented approach. To be able to implement changes in complex software systems, we might need to apply them to model first. However, modeling aspect-oriented change realizations by common means is difficult, so it is appropriate to use a dedicated aspect-oriented approach such as Theme. This way, a change is expressed separately. It may also serve as a concise and precise formal documentation. However, in some cases the model must remain purely object-oriented. Also, with a large number of aspect-oriented change realizations dependencies arise among them, which is very difficult to track. In such cases it is useful to perform a model composition while still keeping the aspect-oriented form of changes for documentation purposes. In this paper algorithms are proposed that describe the process of model composition.

1 Introduction

In order to achieve improved modularity and component reusability of a software system, it would be appropriate to use aspect-oriented approach for realization of change requirements. This way, a change is expressed separately. It may also serve as a concise and precise formal documentation.

Modeling aspect-oriented change realizations is possible with the Theme approach [1]. This approach consists of two parts: Theme/Doc and Theme/UML. Theme/Doc covers identification and analysis of crosscutting themes, while Theme/UML covers designing and composition of themes. In Theme approach, a theme is a collection of structure and behavior that represent one feature. The Theme model distinguishes between two kinds of themes: base themes, which may share some structure and behavior with other base themes, while modeling these from their own perspective, and crosscutting themes, which have behavior that overlays the functionality of the base themes.

* Master study programme in field: Software Engineering
Supervisor: Assoc. Professor Valentino Vranić, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 245–250.

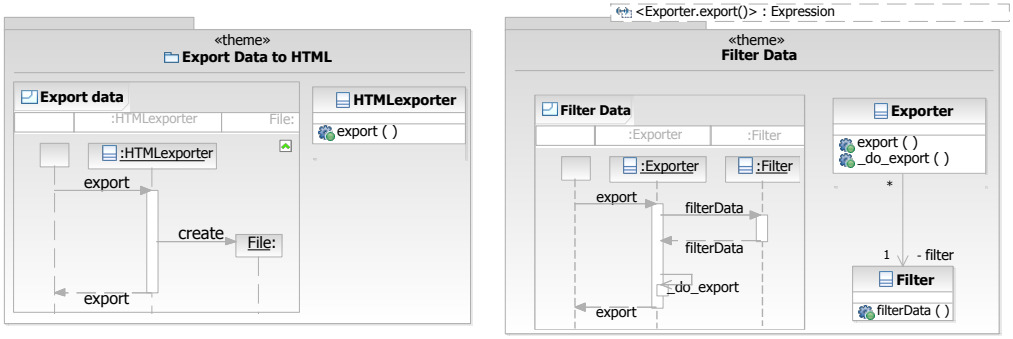
The crosscutting themes are aspects. In Theme/UML, base and crosscutting themes internals are represented mainly by class and sequence diagrams.

However, in some cases the model must remain purely object-oriented. Also, with a large number of aspect-oriented change realizations dependencies arise among them, which is very difficult to track. The composed model would allow to see how would these aspect-oriented change realizations affect original application in the object-oriented model. In such cases it's useful to perform a model composition while still keeping the aspect-oriented form of changes for documentation purposes. In this paper algorithms are proposed that describe the process of model composition.

The rest of the paper is organized as follows. Section 2 describes applying change realization to a model of a real application. In Section 3 the algorithms for model composition are proposed. Section 4 describes evaluation of the composition proposed in this paper. Section 5 presents a comparison to related work. Section 6 presents conclusion and future work.

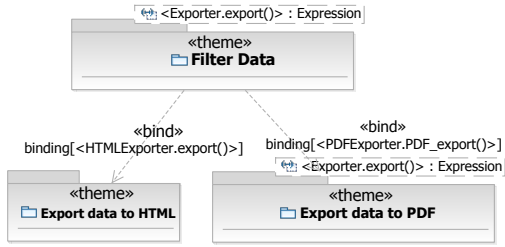
2 Scenario Adaptation Of Application Model Based On Change Requirements

Some change requirements were applied to a real application named Cinema Inviter at model level in the aspect-oriented way. Cinema Inviter is a simple application that parses web content and exports a file that contains a set of information that might be interesting to a user. This information is filtered by a location that a user can choose via application GUI. This application has been developed at model and implementation level. After that, some change requirements were identified and realized at the model level using the Theme approach. For better understanding of aspect-oriented change realizations at the model level with Theme the realization of change requirement "Output data would be filtered by various constraints specified by user" is presented in Figure 1.



(a) Base theme in Theme/UML

(b) Crosscutting theme in Theme/UML



(c) Binding view in Theme/UML

Figure 1. Realization of the change requirement "Output data would be filtered by various constraints specified by user.".

```
foreach class in base theme do
    copy class into composed object-oriented model with all its operations and attributes;
end
foreach class in crosscutting theme do
    if class already exists in composed object-oriented model then
        merge class with existing class;
    else
        copy class into composed object-oriented model with all its operations and attributes;
    end
end
foreach association in base theme do
    copy association into composed object-oriented model;
end
foreach association in crosscutting theme do
    if association already exists in composed object-oriented model then
        continue;
    else
        copy association into composed object-oriented model;
    end
end
```

Figure 2. The algorithm of the composition of a class diagram in a base theme with a class diagram in a crosscutting theme.

A base theme describes a module in the original object-oriented application model (the module for exporting files in this case). A crosscutting theme describes an aspect-oriented change realization which is then bound to the base theme with the Theme/UML bind construct.

3 Model Composition

The term "model composition" is in this paper used for a technique that achieves merging of the original object-oriented application model with an aspect-oriented model of change realizations. Result of that merging is an object-oriented model of the application composed with change realizations in an object-oriented way.

An object-oriented application model contains several diagrams that could be composed with an aspect-oriented model. There are structural and behavioral diagrams. Theme/UML diagrams contain classes with additional methods and fields, so they can just be moved into bound classes from the object-oriented model. Composing sequence diagrams with Theme/UML diagrams is more complex, but possible.

On the basis of manual model composition performed on an example described this paper, the algorithms for model composition are proposed and expressed in pseudo code. This includes algorithm in Figure 2 of the composition of a class diagram in a Theme/UML base theme with a class diagram in a Theme/UML crosscutting theme and algorithm in Figure 3 of the composition of a sequence diagram in a Theme/UML base theme with a sequence diagram in a Theme/UML crosscutting theme. Following these algorithms enables to perform manual model composition. They can also be helpful in implementing a tool for automatic model composition (out of the scope of this work).

These algorithms are at a high level of abstraction and it would be needed to extend them at the implementation level for a successful completion of the whole process of model composition. For example, algorithm in Figure 2 at line 5 it is not trivial to decide if the class in the crosscutting theme does not exist in the composed object-oriented model or it exists, but under a different name.

```
copy all lifeline entities from base theme into composed object-oriented model;
foreach lifeline entity in crosscutting theme do
    if entity already exists in composed object-oriented model then
        continue;
    else
        copy entity into composed object-oriented model;
    end
end
new list of first message calls;
foreach sequence diagram in crosscutting theme do
    add first message call in diagram into list;
end
foreach message call in sequence diagram in base theme do
    if list of first message calls contains actual message call then
        copy all message calls from related crosscutting theme into composed object-oriented model;
    else
        copy actual message call into composed object-oriented model;
    end
end
```

Figure 3. The algorithm of the composition of a sequence diagram in a base theme with a sequence diagram in a crosscutting theme.

When composing diagrams shown in Figure 1(a) and 1(b), the Exporter class in the aspect-oriented model is bound to the HTMLexporter class from the object-oriented model (see Figure 1(c)) and the PDFExporter class from the aspect-oriented model, so all properties and relationships of the Exporter class should be woven into the HTMLexporter and PDFExporter classes although their names do not match.

After the model composition of the change realization shown in Figure 1 a class named Filter (with all the attributes, operations, and associations related to it) should be added into the composed object-oriented model. However, since the Filter class is associated with the Exporter class in the crosscutting theme, it would be associated with the HTMLexporter and PDFExporter classes in the composed object-oriented model.

4 Evaluation

Figure 4 shows a small part of composed sequence diagram from the model composition presented in this paper. To uninformed persons, it would be very difficult to determine which parts of the model are parts of the original application, and which parts were introduced by change realizations.

Furthermore, if afterwards for some reason it would be required to refactor or remove a particular change realization from the model, the developer would have to first search all incriminated places in the model, and after that to modify the corresponding parts of the model.

In this example all change realizations were made using aspect-oriented change realization with model composition. Therefore, all changes were designed with the Theme approach keeping them separated from the original object-oriented or composed object-oriented application model. In this case, in responding to the above mentioned requirement of refactoring or removing a change from the model without a model composition tool, all incriminated places in model would be possible to track using the corresponding Theme/UML diagrams. For example, Figure 4 contains the Filter class with the filterData() method that has been introduced using aspect-oriented change realization. From the corresponding Theme/UML diagrams that are shown in Figure 1, we can trace this change realization into the composed sequence diagram. Thanks to change introduction using model composition the

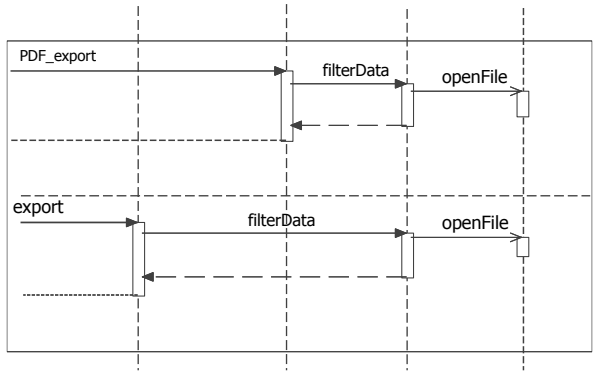


Figure 4. A part of composed sequence diagram.

change realizations are documented concisely and separately from the original application model and are described in principally equal notation as employed in the application model.

Performing the model composition manually exhibits one significant complication that occurred during composition of class diagrams in the example presented in this paper in the composition of the class diagram shown in Figure 5.

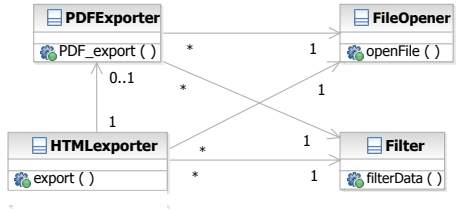


Figure 5. Part of composed class diagram.

When the aspect that introduces new class into model is being composed with the model and this aspect affects some other aspects that were not composed with the model yet, one must not forget to model their relationships after bringing these aspects into the process of composition. During the composition of the FileOpener class that is introduced by an aspect, the PDFExporter class that is affected by this aspect and is also introduced by another aspect by itself haven't been composed with the model yet. Because of this, during the composition of the PDFExporter class the relationship between these two aspects has not been added by mistake, so the composed object-oriented model became incorrect.

An automated model composition might be very useful here. If a change requirement would affect significant part of the application model, it would be very effective to model and manage that change realization by the aspect-oriented approach with automated model composition. Also, the problem with composition of multiple aspects that affect each other would be eliminated.

5 Related Work

Carton et al. [3] aim on integrating Theme/UML with model-driven engineering. Each element that can be involved in a composition is defined by its own metaclass. This metaclass implements an interface that abstracts the notion of a matching criterion. This matching criterion is specific to each element and is implemented in a manner appropriate to the element being matched. Unfortunately, neither implementation details nor the algorithms employed are described.

However, an analysis of the composition metamodel leads to an assumption that Carton et al. are coupling elements into a hierarchy of themes (mapping elements into a composition metamodel) and doing composition on top of that hierarchy. On contrary, the algorithms proposed in this paper perform element mapping incrementally during the composition of each component.

Baudry et al. [2] have developed symmetric composition techniques for composing aspect-oriented models. Symmetric composition is analogous to composition of base themes with each other while this paper focuses on composition of base themes with crosscutting (aspect) themes. Baudry et al. implemented a composition tool in the Kermeta metamodeling language and presented it on an example.

Baudry et al. divide composition implementation into two phases: matching phase and merging phase. However, algorithms presented in this paper were designed at a higher level of abstraction, so they include no matching phase. Baudry et al. developed their merging phase for symmetric composition, so their approach is comparable to merging base themes in this paper. The merging algorithm proposed by Baudry et al. that cover merging class diagrams only is very similar to algorithm for composition of class diagrams proposed in this paper. If all instances of word "crosscutting" would be changed for word "base" in algorithm in Figure 2, it might get the same results as their algorithm.

6 Conclusion and Future Work

Employing aspect-oriented change realization keeps changes separately expressed. It may also serve as a concise and precise formal documentation. However, in some cases the model must remain purely object-oriented. Also, with a large number of aspect-oriented change realizations dependencies arise among them, which is very difficult to track. In such cases it is useful to perform a model composition while still keeping the aspect-oriented form of changes for documentation purposes.

In this paper algorithms are proposed that describe the process of model composition in the Theme/UML approach, while keeping the original aspect-oriented form of changes for documentation purposes. The algorithms have been evaluated on a small study that pointed to error-prone places in them.

Next possible steps could be improving presented algorithms, especially extending them with the matching phase. After that, it would be appropriate to implement the tool for automated model composition and evaluate it on a study.

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/1221/12.

References

[1] Baniassad, E., Clarke, S.: Theme: an approach for aspect-oriented analysis and design, Software Engineering, 2004. ICSE 2004. In: *Proceedings. 26th International Conference*, vol., no., pp. 158-167, 23-28 May 2004

[2] Baudry, B., Fleurey, F., France, R., Ghosh, S., and Reddy, R.: Providing Support for Model Composition in Metamodels. In: *Proceedings of the 11th IEEE International Enterprise Distributed Object Computing Conference (EDOC '07)*, IEEE Computer Society, Washington, DC, USA, p. 253.

[3] Carton, A., Driver, C., Jackson, A., and Clarke, S.: Model-driven theme/UML. In: *Trans. Aspect-Oriented Software Dev*. 2009. VI 5560.

Detection of Code Clones: Necessity or a Myth?

Ján SÚKENÍK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova 3, 842 16 Bratislava, Slovakia
sukenik08@student.fiit.stuba.sk

Abstract. Copy-pasting of source codes is the most popular and bug prone way of code reuse ever performed in an industry. All programmers are told to avoid cloning code and all ignore this advice whenever possible. Is the reason laziness? Would better tools help? Recent research suggests that aggressive refactoring of clones may not be necessary. We present an overview of this research and experiments with clone detectors for Ruby language, including our own tool. We show where they work well and where they fail and discuss the need for the tools that will simplify tracking of code clones evolution.

A paper based in part on this paper was submitted to a peer reviewed scientific journal.

* Master degree study programme in field: Software Engineering
Supervisor: Dr. Peter Lacko, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

Method for Social Programming and Code Review

Michal TOMLEIN*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
michal@tomlein.org

Abstract. Peer code review is a powerful tool, which can be used to significantly improve the quality of programming courses. In our work, we set out to explore a new kind of collaborative programming exercise, where a group of students works on a chain of programming assignments with the goal of their completion by all of the students. To that end, we base our approach on a combination of peer review and social awareness. Students are each assigned a reviewer, who has a live view of their code and can provide feedback by means of the built-in messaging feature. Additionally, we raise students' awareness of their part in the overall progress of the group using a special group progress visualisation. In this paper, we describe the various methods and techniques we employ in order to get the desired engagement of the students and the overall improvement of the learning process. We also describe the outcomes of our initial experiment and outline our plans for future work.

1 Introduction

Code review as a process has seen wide acceptance in the industry as an effective means of ensuring the quality of software. While its application in practice is not equally widespread, it is still very common, especially within companies developing mission-critical software.

However, in addition to being an important quality assurance method, it is also a powerful learning tool. We believe it can effectively be used to significantly improve the outcomes of the learning process and its overall quality. Furthermore, its adoption in the learning process can serve to prepare students for code review in development practice, which is also highly desirable.

In recent years, social approaches to software development have changed the way we look at code sharing, collaboration and the development process. With the advent of social programming and code sharing services such as GitHub, there have been sweeping changes to the way we perceive and expect development of open source software components to work.

Again, though, social programming has not made its way into programming courses to the same extent. In our work, we intend to combine the benefits of both code review and social programming, to improve the quality of the courses to provide additional development skills.

* Master degree study programme in field: Software Engineering
Supervisor: Dr. Jozef Tvarožek, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 257–262.

2 Related work

Code review has been subject of research for several decades. Several different types of review have been studied, ranging from very formal review such as Fagan inspection [4], which involves a multi-stage structured detailed process, to lightweight review such as pair programming.

Studies have demonstrated that there are significant differences in reviewing and debugging skills of people with similar background. A study on 64 undergraduate students on the effects of the Meyers-Briggs personality indicator on debugging skills by Devito Da Cunha and Greathead [2] showed some limited relationship. It was found that "intuitive" users performed better than "sensing" users, more specifically users, whose personality corresponds with thinking in logical connections rather than weighting the options, performed better in debugging tasks.

When viewed more generally, the problem of finding (or recommending) a reviewer is a type of people-to-people recommender, in which reciprocity facilitates successful interactions [5]. In other words, not only does the reviewer need to be good, the reviewer also has to be "compatible" with the user who is requesting help (review).

Peer code review has also been explored within the context of computer supported and collaborative learning, as more and more educators show interest in introducing code review to their courses. Wang et al. [7] have implemented a PCR-based assessment process into a programming course, significantly improving the learning outcomes. They have found that students show acceptance of real-time assessment, are ready to give criticism to their classmates, and are satisfied with this approach overall. Hundhausen et al. [3] compared online vs. face-to-face code reviews and proposed ways for online code reviews to be improved. Tang [6] proposed a distributed social code review tool for programming that improved the overall reviewing process.

Reviewer assignment is a problem that is often also confronted in a different context, especially in reviewing academic papers or research proposals. In these cases, reviewers are typically assigned based on their familiarity with the topic of the research paper.

3 Method description

The primary goal of our work is to design and evaluate a method combining social programming and code review targeting programming courses with the possible generalisation towards the domain of software development.

This method is based on real-time code review using a shared live view of the code being reviewed, which the reviewer can see and comment on. This is in contrast with more traditional code review methods, which are asynchronous and therefore do not happen in real time, nor require the reviewer to be available at the same time as the author of the code.

Secondly, the method incorporates aspects of social programming to create a sense of community within the programming course. The intent is to raise the students' awareness of the whole group's progress and their place within the group and to encourage them to participate in code review to help their fellow classmates complete all of the programming assignments.

3.1 User model

The method relies to a large extent on a user model comprising two important parts, which are both based on the inputs processed by the method, namely:

- User characteristics – to obtain personality and other traits such as learning style,
- User actions – to calculate the reviewing abilities.

We employ a Big Five personality traits questionnaire and an Index of Learning Styles Questionnaire to model the users' personality and characteristics. Once obtained, these traits are typically stable across different assignments. The reviewing abilities, on the other hand, are more variable and tied to specific knowledge domains. Based on user action logs containing information

on successes and failures of previous assignments, including more fine-grained details on the user actions leading to these outcomes, we calculate the reviewing abilities of the user, which are split into two parts:

- The ability to successfully *deliver* help (in the form of a review),
- The ability to *receive* help (a review) and act upon the feedback.

3.2 Reviewer assignment

To enable code review in the context of a programming course, reviewers need to be assigned in a different manner than has traditionally been the case within companies developing software. This is mostly due to the following constraints of a typical programming course seminar:

- Limited time span within which a large number of reviewers need to be assigned.
- Self-assignment is not an option if control is to be retained over collaboration among students.
- Assignment by a supervisor creates a time-consuming distraction from the task of supervising.

For these reasons, there is a need for *automatic reviewer assignment*, which we define as the problem of finding a suitable reviewer for a particular user of the system to help when a problem arises by commenting on the code and suggesting a fix.

We consider reviewers logged into the system and not currently occupied with another review to be available for automatic assignment. Unlike availability, *willingness to help* is more difficult to establish beforehand, but is somewhat possible to estimate based on the outcomes of previous assignments.

In addition, a certain level of *familiarity with the given topic* is required for a particular reviewer assignment to be fruitful. Such familiarity can easily be established by the use of a user model constructed using questionnaires such as the Big Five personality traits or the Index of Learning Styles and using logged actions and attained task scores.

We define the *success of a reviewer assignment* as an improvement over the previous state of the reviewed code as a direct or indirect result of the feedback provided by the assigned reviewer. The exact definition of what qualifies as an improvement of the code is highly dependent on the application domain. In the context of a programming course, this may be a successful compilation or the passing of a unit test.

In order for the method to assign suitable reviewers, it needs to establish the *probability of success of a reviewer assignment*. To model the probability, we employ a Rasch model. The inputs for this model are the users' reviewing abilities, *deliver* and *receive,* introduced above as part of the user model.

Let $deliver_i$ and $receive_i$ denote the reviewing abilities of user U_i and let $Pr(i, j)$ denote the probability of success when user U_i reviews the code of user U_j, then:

$$\Pr(i,j) = \frac{e^{deliver_i - receive_j}}{1 + e^{deliver_i - receive_j}}$$

This implies that when the ability of the reviewer to deliver help through a review matches the ability of the student to receive the help, the expected probability of success is 50 %. The probability of success approaches 100 % for higher values of *deliver* relative to *receive*.

3.3 Two-phase approach

While personality traits can be established beforehand (e.g. using questionnaires), reviewing abilities of the individual users are unknown at the beginning of reviewer assignment. To account for this, the method works in two phases: a *calibration phase* and a *performing phase*. In the

calibration phase, reviewers are assigned randomly in order to gather data on the outcomes of the initial reviewer assignments. This data is then used for maximum likelihood estimation of the reviewing abilities.

Once the determined reviewing abilities have passed a given threshold of measurement error, we are able to correlate the personality traits of the users with their reviewing abilities. This provides default values of reviewing abilities for new users whose personality traits are known. In the performing phase, reviewers are assigned according to the predetermined reviewing abilities.

3.4 Cross-task assignment

In a programming course, students are often assigned the same task or exercise to solve. It is undesirable for them all to collaborate directly on its solution. In this context, the method is expected to enable collaboration through code review only. We see three distinct approaches that can be taken:

- *Intra-Task Assignment* – where a user's reviewer may be working on the same task.
- *Cross-Task Assignment* – where a user's reviewer may only be working on a different task.
- *Task Chain Assignment* – which is a variation of Cross-Task Assignment, where a user may only be assigned a reviewer who has already completed the task the user is working on and has moved on to the next task in the chain.

The advantage of chaining tasks and requiring the completion of a task in the chain in order to be assigned as a reviewer of that task is manifold:

- A reviewer who has completed a task is more likely to be able to help another student with it.
- A reviewer who has completed a task in the chain has nothing to gain from seeing the code of a student working on the task.

This is not true of cross-task assignment without a task chain, even if we add the requirement that in order to be assigned as a reviewer of a task, one needs to have completed that task beforehand. This is because without an ordered sequence or chain of tasks, seeing the code of another student may still influence the reviewer's solution of the task if it is later assigned to the reviewer.

3.5 Time segmentation

In programming courses, students are often under significant time pressure to complete the assigned tasks in time. Because of this, it may be difficult for them to participate in real-time code review, because they would essentially be using their already significantly constrained time to read, understand and provide feedback for someone else's code.

It could be argued that the best feedback will come from students who have solved their assigned tasks with time to spare before the deadline, if for no other reason than that they are the best programmers in class. This assumption, however, does not hold if the student and the reviewer are working on a different task, possibly of different complexity. Especially in the case of a task chain where the complexity of the tasks increases with each task, a reviewer experiencing trouble with task 2 may still be able to help a student working on task 1.

The solution to this problem is to divide the time allotted for the solution of the programming task into segments. There are two kinds of time segments:

- *Code Editing* – when the student can edit their own code, but can also see the code they are assigned to review and provide feedback, and
- *Code Review* – when the student can see their own code, but not edit, and they can see the code they are assigned to review and provide feedback.

However, with a single schedule for all students, it is impossible for students whose code is being reviewed to act upon the feedback they receive during the code review time segment, as code

editing is disabled during this time. A better solution is to divide the students into groups with different time schedules. The division into groups must satisfy the condition that no student-reviewer pair belongs to the same group.

3.6 Social awareness

The context of programming courses presents an opportunity to implement social programming in an innovative manner. The approach we take with the method is based on a task chain.

A task chain is an ordered sequence of tasks with increasing complexity. Students are expected to complete all or as many as possible of the tasks in the chain. In accordance with task chain assignment, a student may only be assigned a reviewer who has already completed the task the student is working on and has, therefore, moved on to the next task in the chain.

We add a social component to the task chain approach by turning the problem of completing the chain into a group effort, so that in order for the chain to be declared completed, every student in the group needs to complete the chain. To make this possible, we visualise the current progress of the group, making it possible for every member to see how they are doing with respect to the rest of the group and how far the group as a whole is from the final goal.

Figure 1 presents an example of a task chain visualisation showing two points in time, first with the majority of students working on task 2, then (at a later point in time) with the majority of students working on task 4. Each time a student completes a task, he or she becomes available as a reviewer for the previous task. For example, in the first scenario in Figure 1, there are 7 students, who can be assigned as reviewers to some of the 11 students working on task 2, and there are 18 potential reviewers for task 1.

Reframing the completion of tasks in terms of a group effort provides motivation to participate in code review. This is especially true if the grading system is set up to reward helping the advancement of the group on par with one's own progress.

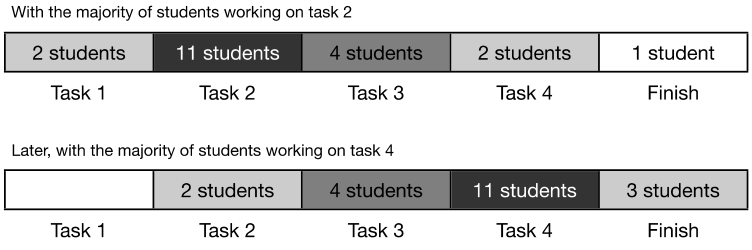


Figure 1. An example of a task chain visualisation showing two points in time.

4 Evaluation

We are currently evaluating the outcomes of the first of the method's two phases. We have collected data on 172 students in an introductory programming course at the faculty. The data contains observations of student work and describes the relationship between the personality traits and characteristics and the reviewing abilities.

We designed the experimental study as a task chain – an ordered sequence of 5 highly interdependent programming assignments on introductory cryptanalysis. The functionality necessary to support the live reviews and random automatic reviewer assignment was built into the existing web-based learning platform deployed within the course, called Peoplia. The following kinds of data are being gathered:

- Reviewer assignment, i.e. which reviewer is assigned to which user and when (approximately 400 random assignments have been logged),
- Messages exchanged among the students (both their content and time sent),

- Compiler output (such as warnings and errors), and test results from the automated tests.

The first batch of data collection is complete. We have constructed a matrix of values assessing the reviewer-to-user assignments and, using Maximum Likelihood Estimation, we have been able to estimate the per-user coefficients for the Rasch model.

Preliminary observations indicate that while students have difficulty communicating with others – they appear to be shy (a personality trait of our sample of undergraduate students) – once they start reviewing each other's work, they end up more successful in solving the problems. Due to the way our initial experiment was structured, this shyness resulted in less activity among the students. However, despite this issue, we are seeing a slight positive correlation between reviewing abilities and the conscientiousness and extroversion values as given by the Big Five questionnaire.

Additional experiments are planned to evaluate the calibrated method in both of its phases. With further results, we expect to be able to determine the relationship between the personality traits of the users and their reviewing abilities with greater confidence and to construct an algorithm capable of applying this knowledge to the problem of assigning a suitable reviewer.

5 Conclusions

The intent of our work is to combine the benefits of code review and social programming. In this paper, we proposed a novel approach to the problem of automatic reviewer assignment. Our goal is to select a suitable reviewer with the highest probability of success in helping the user, based on their ability to deliver and receive help.

We are also in the process of exploring the relationship between the reviewing abilities of students and their personality traits, based on data from our experiments within the course.

By exposing the students to other students' code, we aim to inspire them to improve their own code, their ability to read and understand other code, to learn about different ways of looking at the same problem, and last but not least, to train them to be able to provide feedback, which is in itself an exercise of learning by teaching. Learning by teaching is a well-known practice effective especially in the long term, as it prepares students to learn new concepts later [1].

Acknowledgement: This contribution is the partial result of the Research & Development Operational Programme for the project PerConIK, ITMS 26240220039, co-funded by the ERDF.

References

[1] Biswas, G., Leelawong, K.: Learning by teaching: A new agent paradigm for educational software. *Applied Artificial Intelligence*, (2005), pp. 363–392.

[2] Devito Da Cunha, A., Greathead, D.: Does personality matter? An analysis of code review ability. *Communications of the ACM*, vol. 50, no. 5, (2007), pp. 109–112.

[3] Hundhausen, Ch.D., Agarwal, P., Trevisan, M.: Online vs. face-to-face pedagogical code reviews: an empirical comparison. *Proceedings of the 42nd ACM symposium on Computer science education (SIGCSE'11)*, ACM, New York, NY, USA, (2011), pp. 117–122.

[4] Nidhra, S., Dondeti, J.: Black box and white box testing: A literature review. *International Journal of Embedded Systems and Applications (IJESA)*, vol. 2, no. 2, (2012), pp. 29–50.

[5] Pizzato, L., Rej, T., Akehurst, J., Koprinska, I., Yacef, K., Kay, J.: Recommending people to people: the nature of reciprocal recommenders with a case study in online dating. *User Modeling and User-Adapted Interaction*, (2012), pp. 1–42.

[6] Tang M.: Caesar: A Social Code Review Tool for Programming Education. Massachusetts Institute of Technology. Master's thesis, (2011).

[7] Wang, Y., Li, H., Feng, Y., Jiang, Y., Liu, Y.: Assessment of programming language learning based on peer code review model: Implementation and experience report. *Computers & Education*, vol. 59, no. 2, (2012), pp. 412–422.

Symmetric Aspect-Oriented Programming in JavaScript

Jaroslav BÁLIK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
balki.balkovic@gmail.com

Abstract. Several aspect-oriented programming languages were invented for purposes of seamless implementation of crosscutting concerns although they did not gain popularity and market success. However there exist programming languages that are not intended to solve problem of crosscutting concerns but have features which are very close to aspect orientation. This article explains how can be prototype-based languages like JavaScript used in similar way as languages for multidimensional separation of concerns. Prototype-based programming is object-oriented programing which does not contain classes, just objects and inheritance is achieved by cloning the prototype object and adding desired methods in runtime. This article explains, how to implement subjective objects by prototype object and adding batch of methods.

1 Introduction

Aspect-oriented development is an approach, which intends to enhance modularity of object-oriented programming by creating modules from previously inseparable crosscutting concerns. In spite of provided benefits, the adoption of aspect-oriented languages by industry is very slow. One possible way how to accelerate penetration of aspect-oriented programming to industry is invention of new, and easy to understand programming language. Inventing new programming language is a problematic task and even after releasing language to public, it has no guarantee of wider acceptation. Another possible way is to find established languages, which are not yet denoted as aspect-oriented, and find the existing features which can be used in aspect-oriented implementations. Asymmetric aspect-oriented features can be added to existing programming languages by framework. Some features of programming languages has been identified as analogous to symmetric aspect orientation, these features are open classes, traits, intertype declarations and prototype-based inheritance [1]. Symmetric aspect-orientation is already the part of some programming languages. An important task is to discover the features which can help us to implement programs in symmetric aspect-oriented way. In this article the JavaSctript was chosen to explain symmetric-aspect oriented programming

* Doctoral study programme in field: Software Engineering
Supervisor: Asocc. Professor Valentino Vranić, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 263–268.

by prototype-based inheritance. Section 2 explains the prototype-based programming in brief. Section 3 examines the symmetry of aspect-oriented languages. Section 4 shows symmetric aspect-oriented implementation in Hyper/J. Section 5 shows symmetric aspect-oriented implementation in JavaScript. Section 6 adds some remarks on symmetric aspect-oriented programming in JavaScript.

2 Prototype-Based Programming

Prototype based programming has appeared in 1987 in Self programming language. Prototype-based programming is a type of object-oriented programming, which is diverging from traditional class/object dichotomy. A problem with classes is that class hierarchies are complex, and the classes often play different roles [6], what causes problems with maintainability of application. As a reaction to this complexity Borning [3] has proposed an informal description of classless language in which new objects are created by cloning and modification of prototypes. After cloning, no relation is maintained between the original object and its clone. Besides of cloning, the objects can be created "ex nihilo" by definition of entirely new object.

3 Symmetry of Aspect-Oriented Programming

Symmetry is one of the most important properties of aspect-oriented approaches. In brief, the most popular PARC approach [8] represented by AspectJ language is asymmetric. In AspectJ, the elements of domain code are different from so-called "Aspects". Aspects can affect domain code, or the other aspects. However the elements of domain code cannot affect the Aspects. In symmetric aspect oriented approaches all elements are equal. The most significant symmetric aspect-oriented approach is subject-oriented programming. Main elements of subject-oriented programming languages are subjective classes, which represent the partial view on the whole class. The main representative of subject-oriented programming is Hyper/J programming language [10]. Beside element symmetry, a complex view of symmetry includes relationship and join point symmetry [7].

In the PARC AOP approach, the main decomposition is object-oriented. Crosscutting concerns are encapsulated in elements called aspects. The structure of aspects is different than the structure of the base decomposition elements, which constitutes an element asymmetry. Aspects can affect the base decomposition, but the opposite direction of influence is impossible, so there is also a relationship asymmetry, too.

Subject-oriented programming in Hyper/J implements concerns by the means of partial, subjective classes organized into so-called hyperslices and hypermodules. Subjective classes have the same structure for all concerns, therefore subject-oriented programming is symmetric from the element perspective. Every concern can affect other concerns, so from the relationship perspective, it is also symmetric.

4 Subject Oriented implementation in Hyper/J

Subject-Oriented programming is based on theory of Multidimensional Separation of Concerns [10]. Final object in subject-oriented programming is composed of partial subjective views on object called subjects. In the example, there is an Employee object, which can play two roles. First role is an resident employee, which has his own office in company and certain salary. The second role is an external employee, which is outsourced by the external company and is hired for certain hour tarriff.

Base subjective class contains methods, which are common for all roles.

```
public class Employee{
  private String name;
  private String surname;
```

```
  private int hoursPerWeek;

  public void setName(String name){this.name=name};
  public String getName(){return name};
  public void setSurname(String surname){this.surname = surname};
  public String getSurname(){return surname};
  ...
  }
```

Subjective class for resident employee contains fields for office and hourly wage and methods for manipulation with them.

```
public class Employee1{
  private String office;
  private float hourlyWage;

  private void setOffice(String office){this.office = office};
  private void getOffice(){return office};
  ....
  }
```

Subjective class for external employee contains fields for company and hourly tariff wage and methods for manipulation with them.

```
public class Employee2{
  private String company;
  private float hourTariff;

  private void setCompany(String company){this.company = company};
  private void getCompany(){return company};
  ....
  }
```

Partial classes are composed by composition code. Our application has three concerns Role.base, Role.resident, and Role.external. By equate statement, subjective classes are composed into final class.

```
concerns
class Employee: Role.base,
class Employee1: Role.resident,
class Employee2: Role.external;
hypermodules
hypermodule EmployeeModule: Role.base, Role.resident, Role.external;
relationships:
equate class Role.base.Employee, Role.resident.Employee1 into ResidentEmployee;
equate class Role.base.Employee, Role.external.Employee2 into ExternalEmployee;
mergeByName;
end hypermodule;
```

5 Subject-Oriented Implementation in JavaScript

JavaScript was intentionally chosen for this article as an example of prototype-based language, because of its popularity and wide adoption by web browser companies. Prototype-based programming is one of common paradigms used by JavaScript programmers [5] and can be intentionally used to imitate subject-oriented programming. The theories behind prototype-based programming and subject-oriented programming are independent but several apparent similarities can be found by closer observation. Prototype inheritance in prototype-based programming can be used to implement subjective object in manner similar to subject-oriented programming. In subject-oriented programming, subjective classes are ordinary classes of programming language, which are glued together by glue code. In JavaScript example, the base subjective object is created “ex nihilo”, and other subjective views are added by batches of methods and fields to cloned object.

Prototype-based inheritance lacks powerful tool for quantitative manipulation similar to subject-oriented programming. Composition of subjective views has to be realized per object. Also, compared to subject-oriented programming the prototype-based inheritance cannot produce methods glued together to one method.

In following example, object representing base subjective view “employee” is created.

```
var employee ={
"name":"",
"surname":" ",
"hours_per_week":0,
"getName":function() { return this.name },
"getSurname":function() { return this.surname },
"getHoursPerWeek":function() { return this.hours_per_week },
"setName":function(name) { return this.name = name},
"setSurname":function(surname) { return this.surname = surname},
"setHoursPerWeek":function(hours) { return this.hours_per_week = hours}
}
```

The next code snippet shows the declaration of factory for objects which have employee object as its prototype:

```
var employeeFactory = function(){};
employeeFactory.prototype = employee;
```

Object with role of resident employee, which has defined hourly wage and is accommodated in particular office. The base of object consists of clone of a prototype object. The role, respectively subjective view is added to base object as a batch of atributes and methods.

```
var resident_employee  = new employeeFactory();

resident_employee['office'] ="";
resident_employee['hourly_wage'] = 0;
resident_employee['setOffice'] = function(office){this.office = office };
resident_employee['getOffice'] = function(){return this.office};
resident_employee['setHourlyWage'] = function(wage){this.hourly_wage = wage };
resident_employee['getHourlyWage'] = function(){return this.hourly_wage};
```

Employee in this example can have also a role of external employee. External employee is provided by certain company, certain hour tariff is paid for his work and certain amount of hours is ordered from company.

```
var external_employee  = new employeeFactory();

external_employee['company'] ="";
external_employee['hour_tariff'] = 0;
external_employee['hours_left'] = 0;
resident_employee['setCompany'] =  function(company){this.company = company };
resident_employee['getCompany'] = function(){return this.company};
...
```

6 Capabilies of symmetric aspect-oriented implementation in JavaScript

In example from section 5 a typical additive composition of subjective views is shown. In prototype-based inheritance we can add new methods to the clone of a prototype, but it is also possible to remove unwanted methods. By removing unwanted methods, we can remove entire unwanted concern from object. This option is not yet present in symmetric aspect-oriented programming. In terms of subject-oriented programming it could be "negative" subject, which just removes specified functionality.

In the code example above, the traditional JavaScript prototype inheritance is shown. With help of libraries instead of enhancing cloned objects by adding the batches of methods, it is possible to copy content from one object to another. For example, method JQuery.extend(object1,object2) copies the content of object2 to object1. This approach is closer to original subject-oriented programming and is present in frameworks PrototypeJS and JQuery.

7 Related Work

The Data–Context–Interaction (DCI) [9] paradigm's role based design is very close to symmetric aspect-oriented approach. DCI relies on traits for implementing roles that can be used to emulate symmetric aspect-oriented programming.

Asymmetric aspect-oriented programming is also possible in prototype-based languages in article Aspects in a Prototype-Based Environment [4] the dynamic nature of pointcuts is explained.

Multidimensional separation of concerns, theory behind symmetric aspect-oriented programming is similar to theory of feature-oriented programing [2]. In context of this work, it could be interesting to find unification of multidimensional separation of concerns, feature-oriented programming and prototype-based programming.

8 Conclusion and Further Work

Several programming languages have been denoted as symmetric aspect-oriented. But most of symmetric aspect-oriented languages are still in experimental stage of development. Several well established programming languages provide features, which are similar to symmetric aspect-oriented programming. Theory which is behind symmetric aspect-oriented languages, can be also applied with some constraints to wider field of programming languages. When speaking about JavaScript, there exist vast number of libraries, which are intended to provide enhanced modularity. One possible path of further research is to examine these libraries and create small modifications to enhance their symmetric aspect-orientation.

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/1221/12.

References

[1] Bálik, J., Vranić, V.: Symmetric aspect-orientation: some practical consequences. In: *Proceedings of the 2012 workshop on Next Generation Modularity Approaches for Requirements and Architecture*. NEMARA '12, New York, NY, USA, ACM, 2012, pp. 7–12.

[2] Batory, D., Liu, J., Sarvela, J.N.: Refinements and multi-dimensional separation of concerns. In: *Proceedings of the 9th European software engineering conference held jointly with 11th ACM SIGSOFT international symposium on Foundations of software engineering*. ESEC/FSE-11, New York, NY, USA, ACM, 2003, pp. 48–57.

[3] Borning, A.H.: Classes versus prototypes in object-oriented languages. In: *Proceedings of 1986 ACM Fall joint computer conference*. ACM '86, Los Alamitos, CA, USA, IEEE Computer Society Press, 1986, pp. 36–40.

[4] Cleenewerck, T., Gybels, K., Peeters, A.: Aspects in a Prototype-Based Environment ,Vrije Universiteit Brussel, 2004.

[5] Crockford, D.: *JavaScript: The Good Parts*. O'Reilly Media, Inc., 2008.

[6] Dony, C., Malenfant, J., Bardou, D.: Classifying Prototype-based Programming Languages. In Noble, J., Taivalsaari, A., Moore, I., eds.: *Prototype-Based Object-Oriented Programming: Concepts, Languages and Applications*. Springer, 1999, pp. 17–45.

[7] Harrison, W.H., Ossher, H.L., Tarr, P.L.: Asymmetrically vs. Symmetrically Organized Paradigms for Software Composition. Technical Report RC22685, IBM Research, 2002.

[8] Kiczales, G., et al.: Aspect-Oriented Programming. In Aksit, M., Matsuoka, S., eds.: *Proc. of 11th , ECOOP'97*. LNCS 1241, Jyväskylä, Finland, Springer, 1997.

[9] Reenskaug, T.M.H., Coplien, J.O.: The DCI Architecture: A New Vision of Object-Oriented Programming. `http://www.artima.com/articles/dci_vision.html`, 2009.

[10] Tarr, P., Ossher, H.: *Hyper/J User and Instalation manual*. IBM Research, 2000.

Thread Synchronisation Using Self Modifying Code

Dušan BERNÁT*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
bernat@fiit.stuba.sk

Abstract. Self modifying code, that is a program which changes its instruction to be next executed, appears to be a subject of intensive research in recent years again. As it has been originally used (in the 80s) for implementation of various software copy prevention mechanisms, it is often related to hiding program internals for the sake of intellectual property protection. Another well known use of modifying code concerns polymorphic viruses, which make stealth against pattern matching antivirus scanners.

In this article we discuss absolutely positive, and potentially useful application for multithreaded processes. Some synchronisation patterns can be arranged by rewriting a jump instruction in the code of the other thread. Particularly, in the case of mutual exclusion in two threads regularly accessing common critical section in a loop, this can even lead to zero overhead in the uncontended case. Moreover, the mechanism is based only on atomicity of common memory transfer (move) and does not require CPU to provide any special instruction (compare and swap or similar). Furthermore, we show the possibilities of synchronisation of more than two threads.

1 Introduction

Synchronisation of concurrently running threads is a typical problem in parallel programming. The aim of synchronisation is to arrange particular sequence in execution of independent threads. This mechanism allows for example to maintain consistency of shared data. There are several approaches to achieve thread synchronisation. All usual methods are based on sharing some common variable, which has a semantics of *lock*, *semaphore* or similar. In this article we aim on the possibility of implementation of synchronisation mechanism which is not based on sharing common variable, but rather it is based on changing the program instructions.

Self modifying code has been used as early as in the eighties, mainly for providing program copy protection. When program changes the instructions which are going to be executed, the readability and comprehensibility of the code decreases. This effect was also used by a so called polymorphic

* Doctoral study programme in field: Applied Informatics
Supervisor: Assoc. Professor Pavel Čičák, Institute of Computer Systems and Networks, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 269–274.

viruses, which generated harmful code during run time, thus effectively hiding it against pattern matching based antivirus software. In recent years the research in this area attracts more attention again. The application is usually aimed to hiding real purpose of the code, for the sake of intellectual property protection. Some reference on the subject can be found in [3], [1] or [7].

Although the *von Neumann* (or Princeton) architecture, which is widely spread in today's computer systems, does not make distinction between data and instructions stored in the memory, modern operating systems usually protect pages with executable code from writing (with the support from hardware). Some other tools, for example debuggers, often assumes that memory containing a program is constant. Throughout the article we assume an architecture compatible with Intel `x86` processors. We will illustrate the use of synchronisation in the problem of data equalising among threads, so called *work stealing*.

2 Mutual Exclusion Problem

Mutual exclusion is a typical problem in parallel programming. The aim is to achieve, that the part of the code where two processes (or threads) accesses common shared resource – so called *critical section* – is executed in each time instant at most by one of them. Achieving mutual exclusion has a great importance for consistency of shared data as well as for many other situations. The correct solution to this problem is usually confined by several constraints; the process, which is not in the critical section, must not prevent others to enter critical section, and the next ones are prevention of deadlock and livelock. There are several known synchronisation mechanisms, which solve the problem.

2.1 A standard solution

Thread synchronisation can be achieved in several ways. Purely software solution was firstly introduced by Peterson [4]. More simple (shorter) code can be achieved by the use of special instructions, which atomically execute several operations. These includes instructions like *fetch and add*, *test and set*, *compare and swap* or *exchange*.

But program solutions, tought correct, suffer from *busy waiting* and are not tractable in real applications due to dramatic decrease of overall system performance. Better results could be achieved only with the support from operating system, or architecture. Disabling interrupts solves the problem of mutual exclusion, but it also has some disadvantages (stops hardware interrupt handling, hangs whole system in case of failure), thus it is usually used only in the operating system kernel in order to implement higher level synchronisation mechanisms.

Semaphore, as a general purpose synchronisation mechanism, fulfils all requirements for correct solution of mutual exclusion. It was invented by Dijkstra in 1965 [2]. Its implementation requires some support from operating system to allow sleep and wake of process, thus avoiding a busy waiting. In modern Unix operating systems there are several synchronisation mechanisms available. Processes can use older interface of System V semaphores, or new POSIX semaphores. For threads there are mutexes, conditional variables and other, available in the `pthread.h` library.

2.2 Work stealing problem

The way of load equalising among threads (or processes), when the thread with no data to process takes the data from another one, is called *work-stealing*. In the point, where one thread needs to move data from another one, the synchronisation is needed. Basic principle, as well as some modifications, together with comparison of asymptotic behaviour can be found in [5].

Usual implementation of mutual exclusion via mutexes for work-stealing problem is shown in the Figure 1. Two processes in the loop accesses a critical section. The critical section is whole program section where it accesses its local data so that the other process has to wait, if it needs to access these data too. This solution works reliably, but it requires two function calls in each loop

```
P () {
  while (1) {
    mutex_lock (mtxP);
    if (is_empty (data_P)) {
      mutex_unlock (mtxP);
      mutex_lock (mtxQ);
      move (data_P, data_Q);
      mutex_unlock (mtxQ);
      continue;
    }
    do_work (data_P);
    mutex_unlock (mtxP);
  }
}
```

```
Q () {
  while (1) {
    mutex_lock (mtxQ);
    if (is_empty (data_Q)) {
      mutex_unlock (mtxQ);
      mutex_lock (mtxP);
      move (data_Q, data_P);
      mutex_unlock (mtxP);
      continue;
    }
    do_work (data_Q);
    mutex_unlock (mtxQ);
  }
}
```

Figure 1. Synchronisation using mutex.

```
P () {
  while (1) {
    if (is_empty (data_P)) {
      l=1;
      sem_wait (semP);
      continue;
    }
    do_work (data_P);
    if (l) {
      move (data_P, data_Q);
      l=0;
      sem_post (semQ);
    }
  }
}
```

```
Q () {
  while (1) {
    if (is_empty (data_Q)) {
      l=1;
      sem_wait (semQ);
      continue;
    }
    do_work (data_Q);
    if (l) {
      move (data_Q, data_P);
      l=0;
      sem_post (semP);
    }
  }
}
```

Figure 2. Efficient way using shared variable.

execution (in both processes) even in the case when no process is in the critical section nor tries to enter. This introduces substantial overhead. Moreover, the critical section is quite large, because it spans most of the loop body.

Better solution is shown in Figure 2. This requires in each cycle only one test of shared variable value, which is compared to two function calls significant improvement. The main difference is, that the thread which has no more data to process and needs to acquire it from another thread, does not copy the data itself. On the contrary, it blocks its own execution (sleep) and sets the value of shared variable in order to ask the other thread to move required data. This guarantees, that the code in critical section is executed only by one of the threads.

2.3 Self modifying code

The code in Figure 2 is also base for the solution exploiting the modification of instructions. This enables further reduction of overhead, even as low as zero, in the uncontended case, when the other thread does not try to enter critical section either. In this case the instruction testing the value of shared variable can be replaced by the unconditional jump to beginning of the loop. In the case that

```
P () {
  while (1) {
loopP:
    if (is_empty (data_P)) {
      set (rwQ, ``jmp 0'');
      sem_wait (semP);
      continue;
    }
    do_work (data_P);
rwP:
    asm (``jmp loopP'');
    {
      move (data_P, data_Q);
      set (rwP, ``jmp loopP'');
      sem_post (semQ);
    }
  }
}
```

```
Q () {
  while (1) {
loopQ:
    if (is_empty (data_Q)) {
      set (rwP, ``jmp 0'');
      sem_wait (semQ);
      continue;
    }
    do_work (data_Q);
rwQ:
    asm (``jmp loopQ'');
    {
      move (data_Q, data_P);
      set (rwQ, ``jmp loopQ'');
      sem_post (semP);
    }
  }
}
```

Figure 3. Synchronisation using selfmodifying code.

the other thread needs some data from the first one, it writes to this place the instruction to jump to a different address, where is the program performing the data move. The thread, which required a data, is meanwhile sleeping in the usual way on a semaphore (caused by a `sem_wait ()` call).

This means, that in uncontended case each thread executes the loop without any additional synchronisation functions or tests. The overhead is thus minimal. In the case, that the thread wants to allow another one to access its local data, it sleeps itself (however, this requires a system call) and changes the last instruction of the loop in the other thread from `jmp loop` to `jmp 0`, so the code which is otherwise inaccessible will be executed. When the thread finishes this extra block manipulating the data of another thread, it means leaves the critical section, it restores the last instruction of the loop to be `jmp loop` again and wakes the other thread by calling `sem_post ()`.

The only instruction which must be executed atomically is the one writing the jump opcode into the memory. If the loop is sufficiently small, it is possible to use short unconditional jump, which has on x86 processors one byte opcode `0xEB`. As we can move 32 bits value by one `MOV` instruction, it is easy to change of jump instruction without possibility to interrupt.

2.4 Notes on implementation

Majority of operating systems today try to protect program from changes during execution and does not allow writing to the program memory. Program is stored in the pages which are by the help of underlying architecture marked as read only. Therefore it is necessary, that process changes the level of protection via `mprotect ()` call to allow writing `PROT_WRITE`, before the memory pages containing instructions may change. Otherwise the attempt to write into the code segment causes the protection fault, yielding the `SIGSEGV` signal to be delivered to the calling process by the kernel.

If the execution of calling thread is interrupted by an OS scheduler immediately after the shared variable is set, but in the same time before the `sem_wait ()` operation is executed, it would be possible that the second thread executes whole block following the test of shared variable value. In this case, the `sem_post ()` would be executed even before `sem_wait ()`. If mutex is used instead of semaphore, this situation leads to unlocking the mutex before it has been locked. This is the reason to use semaphore here.

```
label1:  mov $1, x
         jmp end
label2:  mov $2, x
         jmp end
             dots
labelN:  mov $N, x

end:
```

Figure 4. Identifying calling thread by variable x.

To prevent a situation, when one of the threads enters the loop before the other one allowed writing to its instructions, additional synchronisation must be used before the loop code begins. It ensures, that both threads starts to execute the loop with possible changes to the instructions, only when the change of access privileges for code segment has been done.

The address of instruction to be changed can be acquired as an address of label. To make pointer from a label, we can use the syntax extension supported by the `GCC` compiler – so called GNU C extensions, particularly *labels as values* [6].

2.5 Solution characteristics

The method just described above requires to know the code (opcode) of the jump instruction. Therefore it is architecture dependent. This disadvantage is typical, but not specific for the self modifying code. Also standard implementations of synchronisation functions, for instance `pthread_mutex_lock ()` from the `pthread.h` library, is dependant on particular architecture. The efficient implementation which can be executed in uncontended case completely in user space without any system call (`futex` – Fast userspace mutex), requires the use of a special instruction.

The need to know the instruction codes (for jumps) is, of course, quite uncomfortable for the programmer. But these low level details can be hidden via appropriate library.

Write access to the code segment may be considered as a disadvantage too. The change of program usually means a security risk. On the other hand, from the viewpoint of architecture, particularly the Princeton architecture which is widely used in current computer systems, the memory of program and data is equal. Change of the process instructions during its execution is for the architecture as natural as change of the data. However for the programmer it might look unusual.

Properties mentioned above may look on the first sight as disadvantages. Actually these properties are present also in other approaches. A real disadvantage may appear as a negative influence of code segment changes to the instruction cache of processor, as its efficiency is mostly consequence of read only access. Due to modifications in the code of each thread, it is not possible to share the pages with code as usually. This can lead to a slight increase in overall size of required memory.

3 Synchronisation Of More Than Two Threads

In the case that a thread which has no data to process is allowed to chose from more threads to acquire data, we can proceed analogously. The difference is that the thread whose control flow was changed, must find out which thread initiated this change by rewriting its instructions and requires the data. This can be achieved so that each thread will change the jump instruction to a distinct address. The program on particular target address sets the value of a variable so it identifies the thread which initiated the change in control, see Figure 4. These few move and jump lines replaces the `asm` line from Figure 3.

When there are several threads, it is also necessary to treat the situation, when more threads require a data from the same “victim”. In this case it would be possible, that the jump instruction of

the victim's code will be rewritten repeatedly, while the target thread could see only the last change. Apart from other unintended actions, the thread which changed the jump instruction as first would be never woken up (because only the last thread would be served and woken). This can be resolved by using shared array instead of shared variable. The array will have one value per thread. The thread requesting a data must check by the use of atomic exchange instruction that the target thread is free. Otherwise the thread must chose another victim and the procedure repeats.

4 Conclusions and Future Work

In this article we showed a method of thread synchronisation used to solve a work-stealing problem, which is not based on sharing some common variable, but rather it is based on changing instructions in shared code segment. It presents an interesting application of self modifying code, which may decrease the overhead of synchronisation in uncontended case to the minimal possible value.

Both methods, two threads, as well as many threads case, were implemented and successfully approved. Remaining questions comprise quantitative evaluation of the methods and what is the real decrease in time overhead. For more convenient work with the run time changing code, it would be appropriate to implement a library offering the synchronisation functions.

Acknowledgement: This work was supported by Slovak Science Grant Agency VEGA, projects No. 1/0722/12.

References

[1] Anckaert, B., Madou, M., De Bosschere, K.: A model for self-modifying code. In: *Proceedings of the 8th international conference on Information hiding*. IH'06, Berlin, Heidelberg, Springer-Verlag, 2007, pp. 232–248.

[2] Dijkstra, E.W.: The origin of concurrent programming. Springer-Verlag New York, Inc., New York, NY, USA, 2002, pp. 65–138.

[3] Mavrogiannopoulos, N., Kisserli, N., Preneel, B.: A taxonomy of self-modifying code for obfuscation. *Computers & Security*, 2011, vol. 30, no. 8, pp. 679–691.

[4] Peterson, G.L.: Myths About the Mutual Exclusion Problem. *Inf. Process. Lett.*, 1981, vol. 12, no. 3, pp. 115–116.

[5] Plachetka, T.: Clairvoyance versus cooperation in scheduling of independent tasks. In: *ITAT*, 2010, pp. 39–46.

[6] Stallman, R.: 6.3 Labels as Values. In: *C Extensions - Using the GNU Compiler Collection (GCC)*, Free Software Foundation, Inc., 2013.

[7] Tschudin, C., Yamamoto, L.: Harnessing self-modifying code for resilient software. In: *Proceedings of the Second international conference on Radical Agent Concepts: innovative Concepts for Autonomic and Agent-Based Systems*. WRAC'05, Berlin, Heidelberg, Springer-Verlag, 2006, pp. 197–204.

Activity-Based Programmer's Knowledge Model for Personalized Search in Source Code

Eduard KURIC*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
kuric@fiit.stuba.sk

Abstract. Code search engines help programmers to find and reuse software components. To support search-driven development it is not sufficient to implement a "mere" full text search over a base of source code. When a programmer reuses source code he has to trust the work of external programmers that are unknown to him. Therefore, there is desirable to calculate a „karma" value for each programmer. Reputation ranking can be a plausible way to rank source code results. It can be supported by using an externalized model of each programmer's knowledge of a particular source code. In this paper, we propose programmer's knowledge model and methods for its automatic retrieving.

A paper based in part on this paper was accepted for publication in 40th Int. Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM 2014).

* Doctoral degree study programme in field: Software Engineering
Supervisor: Professor Mária Bieliková, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

Modular Operating System

Martin VOJTKO*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
martin.vojtko@fiit.stuba.sk

Abstract. This paper summarizes the proposal and implementation results of a Modular Operating System. The complexity and heterogeneousness of embedded systems is growing. This fact results into problems with compatibility, portability and energy dissipation of developed application software and customization of operating systems. The mission of operating systems is to hide complexity of hardware and to manage system devices. We are presenting a novel concept of an embedded operating system with emphasis on modularity, portability and energy efficiency. This concept leads to reduced time of operating system customization and of application development.

1 Introduction

Complexity, heterogeneousness and energy dissipation of embedded systems is growing. According to Moore's law each 24 or 36 months the number of transistors doubles [5]. This causes growing complexity and energy dissipation of embedded systems. The manufacturers are continuously developing novel hardware to compete against the competitors. This development leads to high heterogeneousness of embedded systems.

Complexity and heterogeneousness increases the time needed for integration of software into new embedded systems. The reusing of software is harder and software adaptation is more time consuming. If an operating system is used in new platform, the problem of software reusing is reduced, but the operating system must be adapted to the new platform. The adaptation time depends on the type of platform change. The change within one family of processors is simpler than the change between families of processors. We can affirm that the architecture of contemporary embedded operating systems must be revised.

The standard operating system is developed to reduce complexity of the processor and its peripheries [7]. The center of each operating system is the kernel. The standard kernel of OS in embedded systems manages the tasks, system memory and I/O devices. Special types of kernels, named micro-kernels, manage only tasks and system memory [4]. In this work we propose a revision of the kernel concept. If we analyze code of the kernel, we can find there parts of code that are platform-dependent and platform-independent. This fact results into organization of kernel into two

* Doctoral study programme in field: Applied Informatics
Supervisor: Assoc. Professor Tibor Krajčovič, Institute of Computer Systems and Networks, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 283–290.

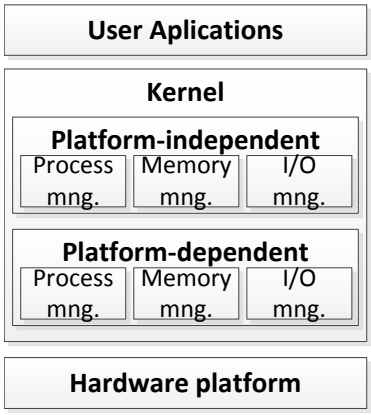


Figure 1. Structure of the embedded operating system kernel [4].

layers (see Figure 1). The division of the kernel increases code reusing, which simplifies the transport from one platform to another. It is not necessary to change the platform-independent code, which results into no or minimal changes in the application software.

As we mentioned, the variety of systems is wide. Besides the portability of operating systems there are other attributes to take into consideration in order to simplify the transfer to other platforms. There is the need for modular architecture of operating systems. Operating systems should consist of modules that vary in energy effectiveness, performance, memory footprint etc. We see OS as a simple frame, which we can fill up by chosen modules which best fit the architecture of concrete embedded system. So we propose that OS should not be only one implementation, but it should be a system of services and packages, from which the consumer can choose.

At present the energy dissipation is one of the most important indicators of embedded systems quality. When comparing two embedded systems, where the first one has lower initial cost price but higher energy dissipation and the second one has lower energy dissipation but worse other parameters, then it is wise to choose the second embedded system. The initial price of the second system might be higher but in the long run this alternative will prove to be more economical. Therefore there is need for the Power management of whole system. The Power management can be implemented as a module to the operating system. As we mentioned, the OS manages the resources. This role can also be extended to power management.

In this paper we present concepts of the Modular Operating System and we provide a brief summary of its architecture. This paper is presenting some related embedded operating systems. We also present results from the testing of the MOS. We compare the results of these tests to test results of a related operating system.

2 Related Work

The release of a new generation of embedded operating systems is near so this part of research is lucrative for many researchers. In this paper we include a brief summary of three embedded operating systems from the research. The first one and the second one are oriented on portability and third one is oriented on energy saving.

The TinyOS offers an interesting concept of operating system. This operating system is build on the concept of three hardware abstraction layers. These layers are connected by interfaces. The bottom layer (Hardware Presentation Layer, HPL) presents the services of hardware to the middle layer. This layer is platform-dependent. The middle layer (Hardware Adaptation Layer, HAL) encapsulates the services of HPL to the top layer. The top layer (Hardware Interface Layer, HIL)

encapsulates the bottom layers into the unified interface. Above the HIL is a layer of platform-independent user applications, which do not change from platform to platform [8].

The FreeRTOS is a popular embedded operating system. The concept of this system is based on many developed ports of this system on many platforms. The developer of the embedded system can choose the platform port and set-up the basic configuration and then the test system is prepared for use. FreeRTOS community supports new platform port development by very good system documentation [3].

DolphinAPI is developed for Wireless Embedded Systems Powered by Energy Harvesting (WESPEH). These systems have mostly no stable energy source, if any, it is based on energy harvesting and recuperation. There is need for very effective energy management. DolphinAPI is implemented to control the state of power source, the energy usage and to safe as much energy as possible [9].

3 Architecture Of MOS

As we mentioned, the kernel of the Modular Operating System (in short MOS) consists of two layers. The first layer is platform-dependent and consists of pieces of assembly code which encapsulate hardware into a higher abstraction layer. The second layer is platform-independent and encapsulates the platform-dependent layer into a presentation layer (see Figure 1). Other MOS modules and user applications can be found above the kernel (see Figure 2) [10].

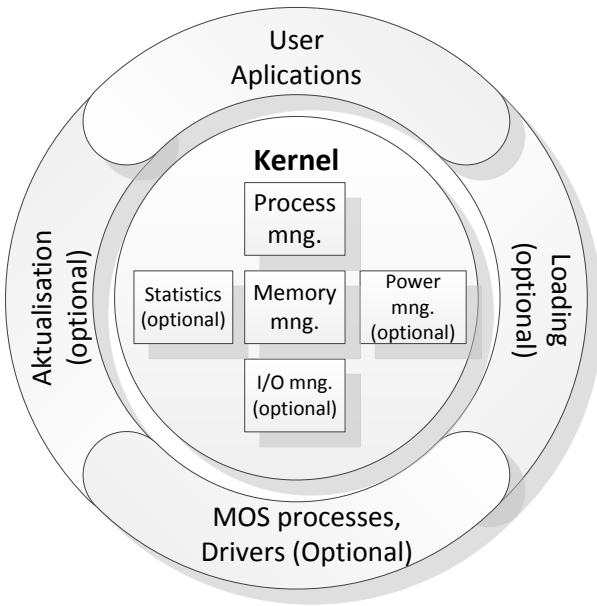


Figure 2. Architecture of MOS [10].

Modules which are presented in Figure 2 can be implemented in more versions. One version can be optimized for best reaction times and another can be optimized for small memory footprint. Each version can be also adapted to more types of platforms. If it is necessary, the modules consist of a platform-dependent and a platform-independent layer.

Compulsory modules of MOS are Process Management and Memory Management. These two modules are the core of MOS and they provide the basic functions of the whole system. It is also possible to use optional modules which do not have to be used (in Figure 2).

3.1 Process Management

The task of Process Management module is to manage tasks running in the system. In order to ensure better change management and portability, the Process Management module is divided into the Task Management sub-module, the Inter-Process Communication (IPC) sub-module and the Scheduler sub-module.

Task Management covers all programs and tasks needs. These needs can be divided into task switch handling, stored programs managing and running tasks managing (creating, destroying, scheduling and inter-task communication).

Programs are stored in the program memory. Each program has its own program header called Program Control Block (PCB). PCB stores information about program memory localization, data size etc. (see Figure 3). PCB is encapsulated in item of list structure for managing purpose.

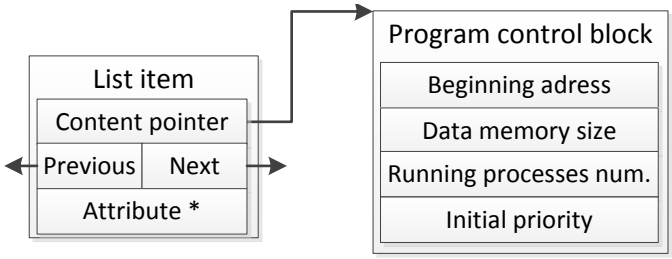


Figure 3. Structure of the Program Control Block and its encapsulating item [10].

Each task has its own header (Task Control Block, TCB) which contains information about the task (see Figure 4). Each TCB is encapsulated in the item structure for scheduling purposes. Platform-dependent part of sub-module includes task-switch which provides extraction of new task context and storing of old task context to its task header or optionally to task stack.

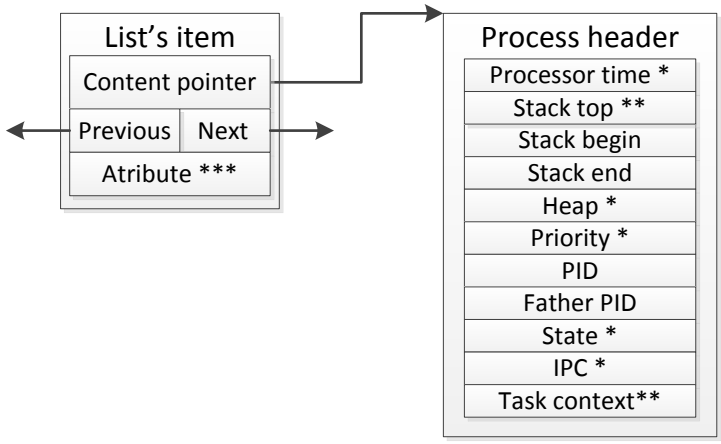


*Figure 4. Task header. * included in header if configured. ** user can choose if the context is stored in the header or in the stack. *** if priority is configured it gets value of this priority [10].*

The Scheduler orders tasks into a waiting list. Task Manager pops the task which will be running in the next cycle from the top of the tasks list.The Task Manager also pushes or inserts the task which was running in the previous cycle. We prepared three types of the schedulers which differ in concept of the tasks ordering:

– *Round-Robin scheduler*, which pushes switched tasks to the list.

– *Priority scheduler*, which inserts tasks into list in an order based on the priority criterion.

– *Multi-list scheduler*, which pushes tasks based on priority into the concrete priority list.

IPC module manages the creation of the communication channels between two tasks. We use Sender-Receiver model for task communication. The sender must register for communication with the chosen receiver. Registration is stored in the list of waiting requests. The receiver pops the requests from list and decides if communication will be created (see Figure 5). Initialized communication is provided by queue data structure.

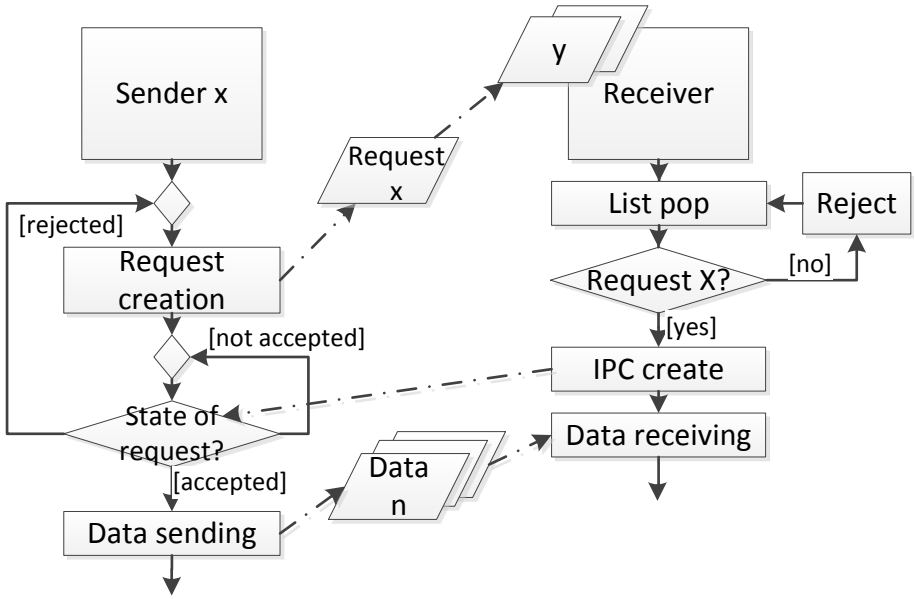


Figure 5. Initialisation and communication between sending and receiving tasks [10].

The strength of modular architecture lies in simple change management and development. User can choose from many suggested models that were implemented by the developer of the OS. We propose to view the OS as a group of many modules in many versions from which the user can combine his own system.

3.2 Memory Management

Memory Management is the second compulsory module. It manages program and data memory of the whole operating system. We decided not to implement a file system for storing data and programs, because it is not important for our research.

Memory can be divided into three partitions. In the first partition the whole program base is stored. The base consists of MOS code and user application task code. MOS is in a state were no changes in program memory can be done until the system is running. We plan to implement a functionality which allows an update or an upload of user applications. For actual research it is not important.

Data created by MOS is stored in the second partition. This partition contains the kernel heap, where the program and task headers and their encapsulation items are stored. This memory is

allocated by the kernel's memory allocate call. This partition also contains the kernel stack where context of kernel procedures is stored.

The third partition is the task heap, which stores data of running tasks. Place for data is allocated by kernel call when the tasks are created and it is freed when the tasks are destroyed. The task heap or kernel heap is organized as a list of memory chunks. At the beginning, the heap contains only one chunk, which has the size of a whole heap (see Figure 6).

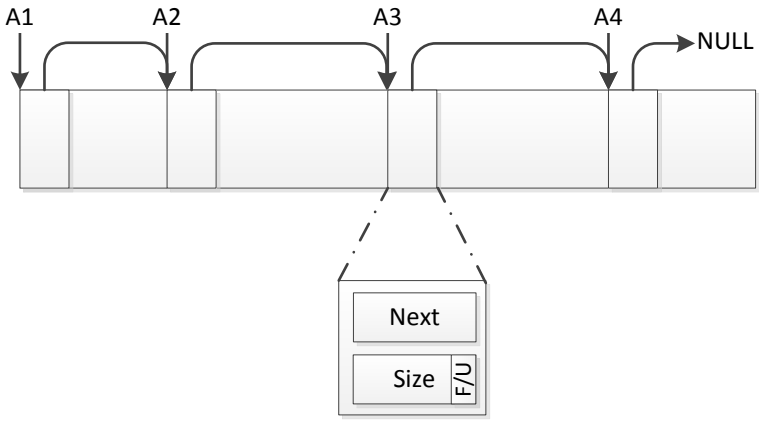


Figure 6. The heap fragmentation process and chunk structure in the heap [10].

Task data is divided into the heap and the stack. The stack contains data pushed by procedure call.The user can manage the heap by memory allocation call. We decided to implement First Fit algorithm for memory allocation. The memory allocation can be simply changed by implementing another algorithm.

3.3 Optional modules

MOS can be set-up as a two-layered micro-kernel but apart from the compulsory modules also other modules can be included in MOS.

We decided to implement a Statistical module for testing purposes. This module collects information from the compulsory modules. This information consists of whole system and individual task running time, program memory usage, kernel memory usage and task memory usage. Statistical module only collects and sends data through the serial port to the host PC where they are processed.

The Power Management module manages energy dissipation of processor and its peripheral and internal devices.The module is using table of devices for optimization.The table contains modes in which the specified device can operate. When task needs a device, system controls if the device is free. In this case the Power Management initializes it. When there is no need for the device any more, Power Management stops operation of device.

4 Results

We implemented port of MOS on processor at91sam7s256 [2] from family arm7tdmi [1]. This processor was embedded in development kit sam7p256 [6]. On this platform we provide analysis of the MOS implementation. The MOS uses $5.5kB$ of program memory at minimal configuration and $9.5kB$ at maximal configuration. This size is comparable to other embedded systems.

Data memory usage is divided into modules (see Table 1). As can be seen data memory usage of MOS is more economical.

Table 1. Data memory usage of system in B [10].

Name of user	Used data in B	FreeRTOS usage in B [3]
Scheduler	$17 + 16$ per task	236
Task minimal	$30 + 16$ item	64
Task maximal	$116 + 16$ item	64
Queue	$25 + 12$ per item	76
List	$25 + 16$ per item	-

At clock rate $48MHz$ and task switch clock rate $10kHz$, task switch takes $14,3\mu s$ which is $14,3\%$ overhead. This time was measured only with usage of Round-Robin scheduler. Other scheduling techniques are dependent on priority and it means that the tasks must be ordered. This process is not deterministic.

For testing purposes, we prepared three programs which were running at mentioned hardware platform together with MOS in two types of task context storage settings and two types of scheduling algorithms. After system start-up and initialization the first task $T1$ is created with priority 7 and then the second task $T2$ is created with priority 4. Task $T1$ also creates task $T3$ with priority 7. Each task is doing simple counting. We were monitoring the work with the system memory and the processing time of tasks (see Table 2).

Table 2. Measured memory usage and processor time distribution [10].

Task	Stack memory used with task context in (B)		Heap memory used with task context in (B)		Processor time in (s)	
	in stack	in TCB	in stack	in TCB	Round-Robin	Priority
MOS	120	56	0	0	$65,7583$	$99,2648$
T0	88	24	$76 + 24$	$76 + 24$	$65,6538$	$49,5437$
T1	120	56	$40 + 16$	$40 + 16$	$0,0903$	$0,0694$
T2	80	36	0	0	$65,7643$	$49,6131$
Kernel	36	36	$624 + 160$	$880 + 160$	$197,2667$	$198,4910$

User can chose where will be the task context stored. If it is stored in the stack, the area of task memory is used more. If the task context is stored in Task Control Block, the area of kernel memory is used more (see columns 2 and 3 of Table 2).

Difference between Round-Robin scheduler and Priority scheduler is shown in column 4 of Table 2. The time distribution is balanced with Round-Robin scheduler. The task $T1$ waits messages from the task $T0$, therefore its processor time is short.

5 Future Work

Research on MOS is not closed. There are many fields where the new OS can be oriented. We plan to improve power efficiency of MOS. This improvement begins with exact analysis of MOS power consumption. Consumption can be measured or modeled in many ways. Analysis of these techniques will be done. The OS can influence power consumption by observing and adjusting the system performance.

We also orient our research on MOS in the sphere of multiprocessor and distributed systems. This decision is promoted by spreading of distributed and multiprocessor embedded systems at world trade.

Our plan is to achieve a complete software platform where user can simply set up Modular Operating System according to his needs, without complicated coding and developing. The result of system set up would be binary code prepared for porting on chosen hardware platform.

6 Conclusions

In this paper we presented the results of research on the embedded operating systems sphere which we named Modular Operating System. Main goals of research were to implement modular, flexible, portable and energy efficient operating system. Experimental OS is still under research but the first results show that the method which we have chosen was right.

Acknowledgement: This work was supported by the Grant No. 1/1105/11 of the Slovak VEGA Grant Agency.

References

[1] Arm-corporation: Technical Reference Manual, ARM7TDMI (Rev 3), 2001.

[2] Atmel-corporation: Datasheet, AT91SAM7S256, 2010.

[3] FreeRTOS-community: The FreeRTOS Project, 2011, http://www.freertos.org/.

[4] Labrosse, J.J., Ganssle, J., Robert Oshana, e.a.: *Embedded Software: Know It All (Newnes Know It All)*. Newnes, 2007.

[5] Moore, G.E.: Cramming more components onto integrated circuits, Reprinted from Electronics, volume 38, number 8, April 19, 1965, pp.114 ff. *Solid-State Circuits Newsletter, IEEE*, 2006, vol. 11, no. 5, pp. 33 –35.

[6] Olimex-Ltd: Datasheet, SAM7-S256 development board, Users Manual, 2008.

[7] Tanenbaum, A.S., Woodhull, A.S.: *Operating Systems Design and Implementation (3rd Edition)*. Prentice Hall, 2006.

[8] TinyOS-community: TinyOS, 2011, http://www.tinyos.net.

[9] Štrba, A.: *Wireless Embedded System Powered by Energy Harvesting*. PhD thesis, Faculty of Informatics and Information Technologies, 2011.

[10] Vojtko, M.: Modulárny operačný systém pre vnorené systémy (in Slovak). Master's thesis, 2012.

Structural Modelling of SOA Design Patterns with Attributed Graphs

Roman ŠELMECI*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
selmeci@fiit.stuba.sk

Abstract. SOA brought significant progress in information systems development. Even though, that SOA brings several advantages, it also brings few disadvantages (e.g. increased difficulty and complex diagrams). One of the promise approaches which refines these disadvantages is application of design patterns. However, these patterns are nowadays published in form which is not understandable by computers. We propose a method based on category theory and attributed graphs for structural modelling of SOA design pattern. In our opinion, this method enables modelling of patterns in computer acceptable form and consequently can support better design quality of SOA based systems.

1 Introduction

The advantages of development based on principles of Service Oriented Architecture (SOA) are beyond a reasonable doubt [5]. However, it also brings some disadvantages. One of them is concerned with the paradigm of Model Driven Development (MDD). By development according to MDD paradigm more types of models are required in order to enable (semi-) automatic generating of final systems. All of these models have to be specified in formal way and we need to define theirs boundaries, entities and relationships among entities. Complexity of these models is also enormous and we need tool support. Another problem is the variability of system models. Each stakeholder in the process can use different notations for description of his/her parts. We face two elementary problems: how to unify different models (or at least find common vocabulary) and how to improve tool support in MDD. Nowadays, there exist several kinds of support of MDD development (e.g. syntax checking or development management), but they rarely offer a mechanism of controlling correctness of whole solution (design). We focus our interest to decrease some aspects of these problems with better application of already gained experiences and tool support.

One concept, which could – according to our opinion – brings promising results in this context, is the concept of design patterns and their utilization in MDD. Design patterns represent natural requirements of every engineering discipline – to reuse some "good practices" of the given

* Doctoral degree study programme in field: Software Engineering
Supervisor: Assoc. Professor Viera Rozinajová, Institute of Informatics and Software Engineering Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 291–298.

field. Patterns are also used in different areas of software engineering – for definition of software structure [7], in MDD as components of platform independent models [9], in code refactoring, or in Domain Specific Modelling (DSM) could be used as basis for Domain Specific Languages (DSL). As well as SOA, design patterns provide independence from implementation technology [6].

The main representative of the design patterns in SOA can be considered patterns published in [5]. This publication contains a lot of patterns (we can find 83 there), which describe together the best design practices in creation of design for SOA based systems. Nowadays these patterns are published in book and consequently used by peoples who apply them – patterns are specified by an informal documentation. Although this specification is useful for people, there exist a lot of limitations by using this kind of pattern specification during development process [1]: (i) users of pattern must be aware of the existence of the pattern and they have to know how to apply it, (ii) seeing that pattern is connected with the context in which it is used, it is highly probable that every user of pattern creates a little different solution, (iii) in case of change of the solution it is difficult to manually modify all areas of the solution which the pattern affected and (iv) manual application of the pattern can involves a mistake into the process of pattern application. Thus, the specification and description of design patterns are critical to their successful application. If we define patterns in computer's applicable form, we could boost up integration of artificial intelligence techniques to model driven development of SOA solutions. Tools could understand this representation of patterns and consequently suggest more correct design of solution or check correctness of existed design.

Our goal in this paper is to propose new approach for structural modelling of SOA Design Patterns [5]. This approach is language independent; supports modelling of SOA based systems and also enable identifying pattern/antipatterns in descriptions/models of developed system. Our approach is based on theory of category, typed graph [4] and object oriented analysis.

The rest of the paper is structured as follows. Related work is given in section 2. In section 3, we describe our approach to structural modelling of SOA design patterns. Evaluation is given in section 4. We conclude with suggestion for future work in section 5.

2 Related Work

Many approaches for partial or entire formal representation of patterns exist in different domains of software engineering. As far as SOA design patterns are concerned, we are not aware of research which would deal with a formal representation of SOA design patterns published in [5].

SOA design patterns are influenced by patterns from other areas. Therefore it can be quite useful to be acquainted with the methods and principles applied for the formal representation of these patterns.

In [10] authors propose a technique for modelling design patterns that allows representing them in suitable way for application's design. Authors analyse two important aspects of design patterns: level of abstraction and level of generality and use meta-schemas for specifying design patterns. Authors in [11] use general mathematical model of design patterns presented in real-time-process algebra. This method supports translation of design pattern specification into code in programming language.

Important role in the environment of the enterprise architecture and the integration of enterprise applications have Enterprise Integration Patterns (EIP) [8]. Authors in [9] were inspired also by these patterns. They expand patterns by configuration parameters in order to create executable enterprise integration patterns. Authors aim to use patterns as platform independent elements in models of integration solution.

Authors in [12] use pattern language for process execution and integration design in SOA and they enhance it by so-called pattern primitives. By means of pattern primitives they want to achieve that the patterns which are described only in informal form could be used in model driven

development. Pattern primitives are so an abstract interface for different participants in solution, which patterns provides. For the purpose of support MDD in SOA they create DSL for every process model. These DSLs are created through meta-models which are based on pattern primitives.

We have investigated the possibility of using some of existing principles for partial formal representation of SOA design patterns. We came to the following conclusion:

Approaches for the object oriented design patterns are mostly specialized on generating code which should be written in the destination programming language or UML diagrams – this approach is not feasible for all SOA design patterns. SOA design patterns often appear in a higher abstract form, which cannot be directly converted into executable programming language code.

In the area of enterprise applications integration we can observe an effort of direct pattern exploitation through the conversion from platform independent model into the executable solutions. This effort is supported also in industry where they start to produce tools which provide pattern implementation from this area. SOA design patterns contrary to EIP are focused more on architecture of system and only few of than can be realized as software tool like many EIP can be.

Approach in [12] establishes formal pattern representation for process integration through meta-models in UML notations. These meta-models are used as bases for domain specific language. This language is however specifically fixated on these patterns and it is not possible to apply it within SOA design patterns.

We examined several approaches from different areas. We noticed that every approach for formalization of design patterns uses well defined structure of patterns. Unfortunately, SOA Design Patterns [5] do not have defined structure in form of connected patterns' participants. However, this structure is essential in many applications of design patterns – refactoring, identifying and modelling. So we need to define their structures.

3 Structural modelling of SOA Design Patterns

If we want to reach better application of SOA design patterns in MDD, we need to find solutions for several problems. Firstly, we can consider about design patterns from different points of view, so it is probably that we will need to use different techniques (e.g. authors in [3] use combination of first-order logic, temporal logic of action and Prolog to formalize structure and behaviour OOP patterns). Other problem is variability of system documentations and models used in SOA. Each one stakeholder in process can use different notation for description of his/her parts. Also for the success of any method there are a few requirements which are important: (1) preciseness, (2) flexibility and (3) tool support.

In our opinion, the most suitable bases for our approach can be found in [2]. Authors propose language-independent and formal approach to pattern-based modeling which could be also used in MDD. Authors of [2] based their approach on graphs, morphism and operations from category theory and exploits triple graphs to annotated model elements with pattern's roles. This approach also support describing (nested) variable submodels, as well as inter-pattern synchronization across several diagrams. Author applied their approach for formalization of object-oriented patterns and propose simple examples how to apply their approach to other kinds of design pattern like workflow patterns and EIP. However, authors did not apply their approach on SOA Design Patterns [5]. We were inspired by these authors in structural modelling of SOA design patterns but we introduced several modifications for utilization of this approach in area of SOA design patterns.

Why we consider that this approach is the most suitable? Rigorous and precise structural modelling of SOA design patterns could be reached with graph and category theories. If we utilize the graph database, we can also bring flexibility and tool support.

In this paper we use examples based on Canonical Protocol Pattern. Table 1 shows profile summary of the Canonical Schema Pattern. In next subsections, our ideas for fundamental structural definition and design pattern profile are proposed.

Table 1: Canonical Protocol Pattern's Profile

Canonical Protocol	
How can services be designed to avoid protocol bridging?	
Problem	Services that support different communication technologies compromise interoperability, limit the quantity of potential consumers, and introduce the need for undesirable protocol bridging measures.
Solution	The architecture establishes a single communications technology as the sole or primary medium by which services can interact.
Application	The communication protocols (including protocol versions) used within a service inventory boundary is standardized for all services.
Impacts	An inventory architecture in which communication protocols are standardized is subject to any limitations imposed by the communications technology.

3.1 Fundamental structural definitions

First of all we introduce same essential definitions from graph theory on which our approach is based. The category *Graph* is a category with objects graphs and graph morphisms. A graph can be described by a tuple $G = (V, E, s, t)$, where V is a set of nodes (or vertices), E is a set of edges, and $s: E \to V$ and $t: E \to V$ are source and target functions, which assign source and target nodes for each edge. Graph homomorphisms are pairs $(m^V, m^E): G^1 \to G^2$ of set morphisms mapping the nodes and edges of the graphs G^1 and G^2, such that the structure of the graph is preserved. The structure of graphs can be enriched with attributes in nodes and edges, as well as with types for nodes and edges. See publication [2] for more details.

We defined design pattern in three levels of abstraction: 1) Design pattern meta-structure, 2) Design pattern structure and 3) Design pattern variant.

Definition 1 (Design pattern meta-structure). Design pattern meta-structure is attributed type graph $DPMS = (TG, Z)$ where TG is an E-graph (see [4]).

Note that DPMS is similar to meta-model definition used in MDD. Figure 1 shows design pattern meta-structure $DPMS = (TG, Z)$ where Z is final DSIG-algebra defined by:

Definition 2 (Design pattern meta-structure data signature). Design pattern meta-structure data signature *DPMSDSIG = String + Pattern participant's vocabulary + Participant relationship's vocabulary*, sorts: Pattern participant's vocabulary, Participant relationship's vocabulary, options see definition 3, and the set of all data sorts used for attribution is $S'_D = \{String,\ Pattern\ participant's\ vocabulary,\ Pattern\ relationship's\ Vocabulary\}$.

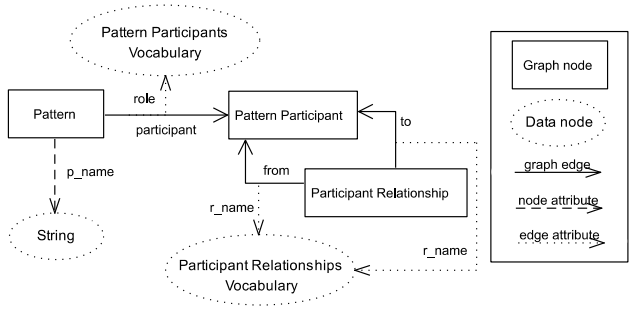


Figure 1. Attributed type graph DPMS = (TG, Z) for pattern's meta-structure.

Definition 3 (Pattern participants' vocabulary and Pattern relationships' vocabulary). The Pattern participants' vocabulary (*PatParVoc*) is finite set of identified objects in informal text pattern specification. The Pattern relationships' vocabulary (*PatRelVoc*) is finite set of identified relationships among participants in informal text pattern specification.

Definition 5 (Design pattern structure). Design pattern structure is an attributed graph DPS typed over DPMS.

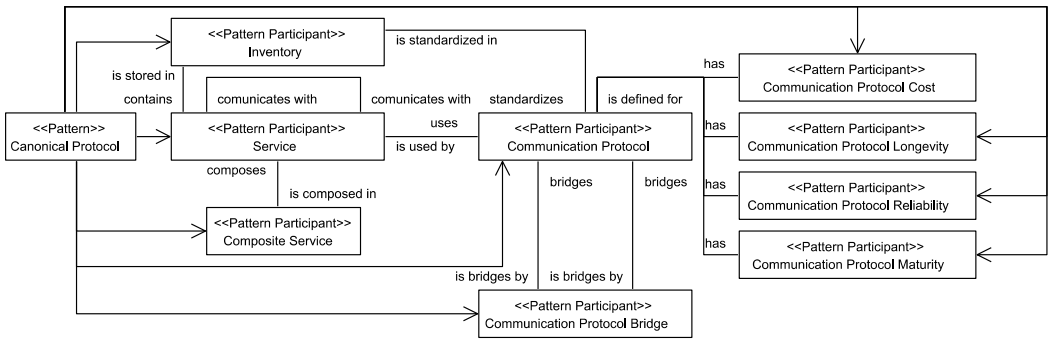


Figure 2. Compact form of an attributed graph DPS for Canonical Protocol Pattern.

The structure of a pattern is usually enriched with information regarding the roles its different elements play in the collaboration, and defining a specialized vocabulary that promotes a common understanding of its parts [2]. Participants and relationships in the design pattern definition are bases for pattern vocabulary.

Definition 6 (Pattern vocabulary). Pattern vocabulary is *PatVoc = (Participant's roles vocabulary; Relationship's roles vocabulary)* where: the participant's roles vocabulary (*PatRolVoc*) is finite set of participants' roles occurred in structure of design pattern. $PatRolVoc \subseteq PatParVoc$ and the relationship's roles vocabulary (*RelRolVoc*) is finite set of participants' relationships occurred in structure of design pattern, where relationship is defined as R = *[source, target]*. $RelRolVoc \subseteq PatRelVoc$.

It is usually that design pattern has several variations. Each one variation represents structure of design pattern in same specific situation of its application. Therefore we need method for capturing these variations. We introduced pattern variant. Pattern variant represents one valid structure in which pattern can exists.

Definition 7 (Design pattern variant). Design pattern variant is attributed type graph *DPV = (TG, Z)* where *TG* is an E-graph (see [4]).

Figure 3 shows design pattern *DPV = (TG, Z)* where Z is final DSIG-algebra defined by:

Definition 8 (Design pattern variant data signature). Design pattern data signature *DPVDSIG = String + Participant's role vocabulary + Relationship's role vocabulary, sorts: Participant's role vocabulary, Relationship's role vocabulary*, options see definition 6, and the set of all data sorts used for attribution is S'_D = *{String, Pattern participant's vocabulary, Pattern relationship's Vocabulary}*.

We also need approach how to describe target model with entities from participant's and relationship's roles vocabularies. We use triple graphs where the source graph is the model of SOA system, the target is the vocabulary of roles, and the correspondence assigns roles to elements in model.

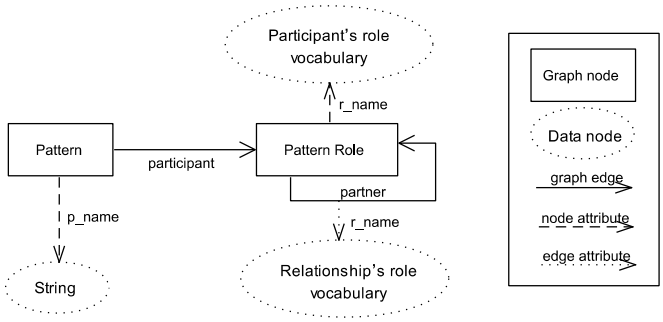


Figure 3. An attributed type graph DP = (TG,Z) for SOA design patterns.

Pattern instance can have many variations. Authors in [7] propose so called *variable pattern* for purpose of capturing variation of design pattern instances. Authors defined variable pattern as:

Definition 9 (Variable Pattern) [2]. A variable pattern is a construct *VP = (P, root, Emb, name, var)*, where:

- $P = \{V_1, ..., V_n\}$ is a finite set of non-empty graphs, where each V_i is called variable part,
- *root* is a distinguished element of *P*, also called the fixed part,
- *Emb* is a set of morphism $v_{ij} : V_i \rightarrow V_j$ with $V_i, V_j \in P$, such that is spans a tree rooted in $root \in P$ with all graphs $Vi \in P$ as nodes and the morphisms $v_{ij} \in Emb$ as edges,
- *name:* $P \rightarrow L$ is an injective function assigning each variable part a name from a set of variables L, of sort $\mathbb{N}$,
- *var* $\subseteq T_{AlgEq}(name(P))$ is a set of equations governing the number of possible instantiations of the variable parts. These equations use variables in $name(P) \subseteq L$, arithmetic operations, and are restricted to use the $<, \leq, =, \geq, >$ relation symbols. Authors call this signature *"Algebraic Inequalities"* and hence T_{AlgEq} (name(P)) is the term-algebra with variables in name(P).

Figure 4 shows simplified graphic notation (based on UML class diagram) of pattern variant for Canonical Protocol Pattern.

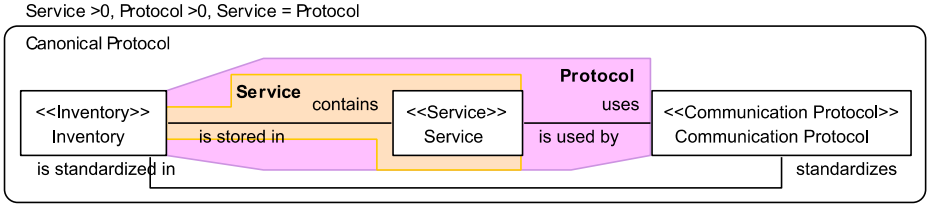


Figure 4. Example of simplified notification of pattern variant definition. Situation when all services in inventory use standardized communication protocol.

3.2 Design Pattern Profile

Design patterns can be specified in different ways. In [1] authors informally specify design patterns by name, problem description, solution description, discussion about pattern impact. Although not all aspects of a design patterns can be formalized, some aspects (structure and behaviour) can be formally specified [3]. We defined design pattern profile as:

Definition 10 (Design pattern profile). A design pattern profile is a *DP = (Name; DPS; PatPro; PatSol; PaCon),* where:

- *Name* is unique name of pattern,
- *DPS* is attributed graph of design pattern structure,

and

Definition 11 (Pattern problem). *Pattern problem (PatPro)* is a finite set of pattern variants which describe possible structures of pattern's participants for problem definition. $PatPro = \{PatProV_1 \dots PatProV_i\}$, where $PatProPV_i$ is pattern variant and $i \in \mathbb{N}$.

Definition 12 (Pattern solution). *Pattern solution (PatSol)* is a finite set of pattern variants which describe possible structures of pattern's participants for solution definition. $PatSol = \{PatSolV_1 \dots PatSolV_i\}$, where $PatSolPV_i$ is pattern variant an $i \in \mathbb{N}$.

Definition 13 (Pattern context). *Pattern context (PatCon)* is a pattern variant which describes pre-required structures of pattern's participants which must be fulfilled before pattern can be used.

Why to specify pattern context? Firstly, pattern context can reduce space for detection of pattern variant's instance. Secondly, some pattern can be applied only under specific conditions, i.e. Enterprise Inventory Pattern can be applied only within organization with sufficient resources and widely accepted standards. This pattern profile can be easy extended with other aspects e.g. pattern impacts.

4 Evaluation

We transformed 14 SOA design patterns into our pattern profiles. These created pattern profiles enable us to define new method for identification of SOA design patterns with graph paths and also enable us to define prototype of DSL. This DSL could be used during design phase of defining new SOA based system or for describing of existed system. Eventually, this description is automatically converted into attributed graph in which we could identify patterns/antipatterns.

5 Conclusion

In this paper we proposed our method for structural modelling of SOA design patterns. This method is based on category theory and attributed graph. We brought in also several approaches for formal pattern representation from different areas of software engineering. The paper also contains fundamental definitions for structural modelling of SOA design patterns. Compared to other approach our method support language independent modelling of SOA design patterns and also support identification of pattern's instances with application of graph database. Our approach could be also applied in context of MDD.

In future work, will continue in creation of new profiles for other patterns. We would also like to define graph language and rules which could support even more formal specification of SOA design patterns and we hope that we create pattern language which enables us to automatize some design decisions during design phase or enable automatic correction of existing design.

Figure 5 shows example of situation when *Security* service from project B need to be stored in *Cedar* inventory from project A. Project A was developed according to principles defined in Canonical Protocol pattern. However, we cannot use actual design of *Security* service because it disturbs these principles and we need to redesign it. Rules defined in graph languages are automatically executed, problem is fixed and modifications are highlighted.

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/1221/12.

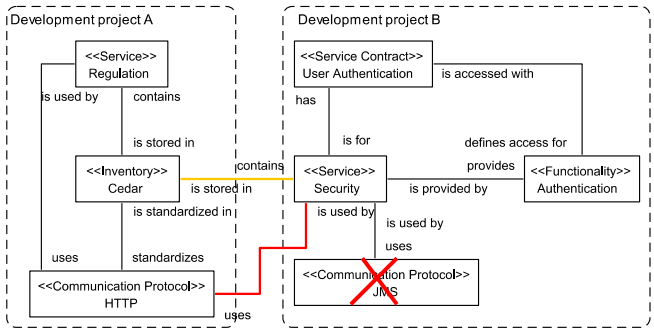


Figure 5. Example of automatic correction of same model's entities according to design pattern rules.

References

[1] Ackerman, L., Gonzalez, C.: *Patterns-Based Engineering: Successfully Delivering Solutions via Patterns*, Addison-Wesley Professional, (2010).

[2] Bottoni, P., Guerra, E., de Lara, J.: A language-independent and formal approach to pattern-based modelling with support for composition and analysis. *Information and Software Technology*. 52, (2010), pp. 821–844.

[3] Dong, J., Alencar, P., Cowan, D.: Formal Specification and Verification of Design Patterns. In: Taibi, T., ed.: *Design Pattern Formalization Techniques*, IGI Global, (2007), pp. 94–108.

[4] Ehrig, H., Ehrig, K., Prange, U., Taentzer, G.: *Fundamentals of Algebraic Graph Transformation (Monographs in Theoretical Computer Science. An EATCS Series)*, Springer-Verlag New York, Inc., (2006).

[5] Erl, T.: *SOA Design Patterns*, Prentice Hall PTR, (2009).

[6] Fowler, M.: *Analysis Patterns: Reusable Object Models*, Addison-Wesley, (1997).

[7] Gamma, E., Helm, R., Johnson, R., Vlissides, J.: *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional, (1994).

[8] Hohpe, G., Woolf, B.: *Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions*. Addison-Wesley Professional, (2003).

[9] Scheibler, T., Leymann, F.: From Modelling to Execution of Enterprise Integration Scenarios: The GENIUS Tool, In: David, K., Geihs, K., Brauer, W., eds.: *Kommunikation in Verteilten Systemen (KiVS)*, Springer Berlin Heidelberg, (2009), pp. 241–252.

[10] Smolárová, M., Návrat, P., Bieliková M.: Abstracting and generalising with design patterns. In: Gürsoy, A., Güdükbay, U., Dayar, T., Gelenbe, E., eds.: *ISCIS'98 – The 13th Int. Symposium on Computer and Information Sciences*, IOS Press, (1998), pp. 551–558.

[11] Wang, Y., Huang, J.: Formal Modeling and Specification of Design Patterns Using RTPA. In: Tiako, P.F., ed.: *Software Applications: Concepts, Methodologies, Tools, and Applications*, IGI Global, (2009), pp. 635–647.

[12] Zdun, U., Dustdar, S.: Model-driven and pattern-based integration of process-driven SOA models. *International Journal of Business Process Integration and Management*. 2 (2007), pp. 109–119.

Computer Networks, Computer Systems and Security

Analysis of Covert Communication via DNS

Timotej TKÁČ*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
timotej.tkac@gmail.com

Abstract. This paper deals with the possibility of communication via DNS. This is a non-standard use of DNS as it primarily translates domain names. The existence of network channel may therefore be obscure. It can be used for example as covert channel between devices infected with malware. Therefore, it is appropriate to analyse different ways of sending and receiving data through the channel. Based on the analysis of problem area and existing solutions, we proposed implementation of software system using different methods of sending and receiving data through the channel. Testing the system on real servers evaluates performance and reliability of the channel.

1 Introduction

Covert communication in network may occur without awareness of other users or administrators. It can be assumed that the content of the hidden messages is malicious and unwanted. In securing the network against data leakage and malware programs, it is appropriate to apply rules to block also communication going non-traditional way. It is not an easy task, since hidden communication is usually not very different from normal ongoing communication that mustn't be blocked. To formulate filtering rules, different ways of entering data into protocols have to be analysed.

In this paper we present a method for storing messages in DNS cache. We focus on storing negative response records in the process of sending messages and checking their presence in the process of receiving messages. Hidden communication via home router or mobile phone with Wi-Fi hotspot feature turned on would be hardly detected. On the other hand, message stored in public name server's cache is accessible worldwide. When choosing server for communication, anycast servers should be avoided, since exact destination cache must be known.

2 Related Work

There are several methods of communication via DNS (e.g. iodine[1], OzymanDNS[2]). One solution is to insert the data into TXT records of the server. This requires administrator access or taking

* Bachelor degree study programme in field: Computer Engineering
Supervisor: Dušan Bernát, Institute of Computer Systems and Networks, Faculty of Informatics and Information Technologies STU in Bratislava

1 http://code.kryo.se/iodine/

2 http://dankaminsky.com/2004/07/29/51/

control over server. Communication via DNS cache does not differ from regular domain name resolution, so public servers can be used.

We have no information of an existing program communicating via DNS cache.

Communication via DNS cache first appeared in [6]. The paper describes how to send, receive and delete data in the memory. Checking the presence of records is done by sending iterative queries. Presence of each byte of the message is indicated by a "start bit". This approach produces redundant queries. Another part of presentation deals with extracting relationships between DNS servers. By analysing TTL values in responses it is possible to calculate time since record was created. The presentation also deals with other methods of communication via DNS that are irrelevant for this work.

More detailed description appeared in [3]. The paper presents more exact description of the channel, with a summary of standard DNS behaviour, introduction of the communication method, benefits and limitations. Solution is adapted to transfer information between computer worms. For checking the presence of memory record worm sends iterative queries.

Another approach deals with measurement of query time [2]. Proposed method was not intended for real data transfer, but as for evaluating DNS server caching properties. The receive operation performs two consecutive queries to read one bit. In our implementation when name server and address space are chosen sensibly, one query per bit is sufficient. Author of the paper also specifies capacity, persistence of information, access time and error rate of such channel.

Results of aforementioned work and standard DNS behaviour [1, 4, 7] are taken into account. Based on their analysis we have proposed different methods of sending and receiving data through the channel

3 Sending Data

The process of sending data is based on sending DNS requests for resolving specific domain names in address space. To represent value of bit 1, the program sends a recursive request for a translation of non-existent domain name. Caching name server stores negative response (with response code NXDOMAIN) obtained from authoritative server into cache and sends it back to the client. Sending bit 0 does not virtually occur, as it is represented as lack of record in cache. Sending is finished after processing all bits of the message.

Negative response message contains SOA record with TTL field. Program sending the data also checks TTL values in all received responses. If the TTL value in the received response is lesser than initial value of a non-existent record, negative record for the queried address existed in the server memory before sending the query. Overlapping address space would cause misinterpretation of the message, thus in this case sending is aborted and new address space definition is required.

TTL values might also be used to determine the time after which it is necessary to send the message again to be permanently available.

Time required for sending a message can be significantly reduced by resolving domain names simultaneously in several threads.

4 Receiving Data

Main purpose of program receiving messages is to decide whether record corresponding to domain name is present in caching server. If a record exists, current address represents value of bit 1. Opposite result represents value of bit 0. Program resolves all domain names of current address space and puts it in correct order to obtain requested message. Based on the analysis of caching servers and principles used in existing solutions, we proposed three different methods of checking presence of records in memory.

4.1 Reading the TTL value of the SOA record

Negative response is stored in cache using SOA record. Keeping this record for a long time would cause a loss of the ability to respond to record change caused by registering a new domain. Even repeated requests for address resolution would be answered by the record stored in cache. Therefore, the period of validity of each negative record in the cache is limited by time in the TTL field of the SOA record. Initial TTL value is configured on the authoritative DNS server for the respective domain. Caching server also limits maximum negative record validity. DNS administrators usually limit negative cache record TTL to 1 or 3 hours.

Server must include SOA record with initialized TTL value in negative response. Using advanced program (e.g. *dig*), the client can display the content of this field.

Time t that has elapsed since the creation of the record in cache is calculated by the following formula:

$$t = TTL_{initial} - TTL_{current} \quad (1)$$

where $TTL_{initial}$ represents value that is used when creating a new negative record in the cache and $TTL_{current}$ is the current value of remaining validity that server included in negative response.

Existing solutions indicate the possibility of preventing access to the stored message by writing the bit value 1 into entire address space (sending recursive queries). Unlike in following methods, exact time of creation of the record allows reading messages deleted this way.

This type of reading cannot be performed on servers that perform NXDOMAIN redirection [5], since SOA record is not present in hijacked responses.

4.2 Query time measurement

Advantage of using a DNS cache is reducing the time required for resolving. Provided cache contains requested record, other servers are not contacted and response is sent in a while. Otherwise server need to send a request to one or more servers. This causes increase in resolution time. By measuring the time between sending a query and receiving a response, we can determine whether the record is present in the cache.

To reduce the effect of variable transmit time between the client and the server, it is recommended to use DNS server with short delay. However, domain names in address space should belong to slow remote server. The more query times of those servers differ, the lower error rate is. Faster resolved domain names represent bit value 1 and slower ones represent value of bit 0. To increase reliability it is suitable to adjust threshold value before and also during receiving process.

This method does not support repetitive message receiving, since recursive queries cause creating records for queried domain names.

4.3 Iterative request response code

If the sender of the request requires iterative translation, the server resolves domain name using only its own records in memory and does not contact other servers. It can be used to detect the presence of records in caching server.

Response indicating the occurrence of the record contains response code *NXDOMAIN*. If the record is not present in memory, the server does not provide a definitive answer but only provides a list of servers that are recommended to contact (NS records) and sets response code to *NOERROR*.

Method of reading the message sending iterative queries is useful when communication takes place across several clients. The message remains in its original version after it is read as iterative client queries do not cause any changes.

This method of reading messages cannot be used on servers that do not support iterative queries processing. This behaviour is indicated by response code *REFUSED*.

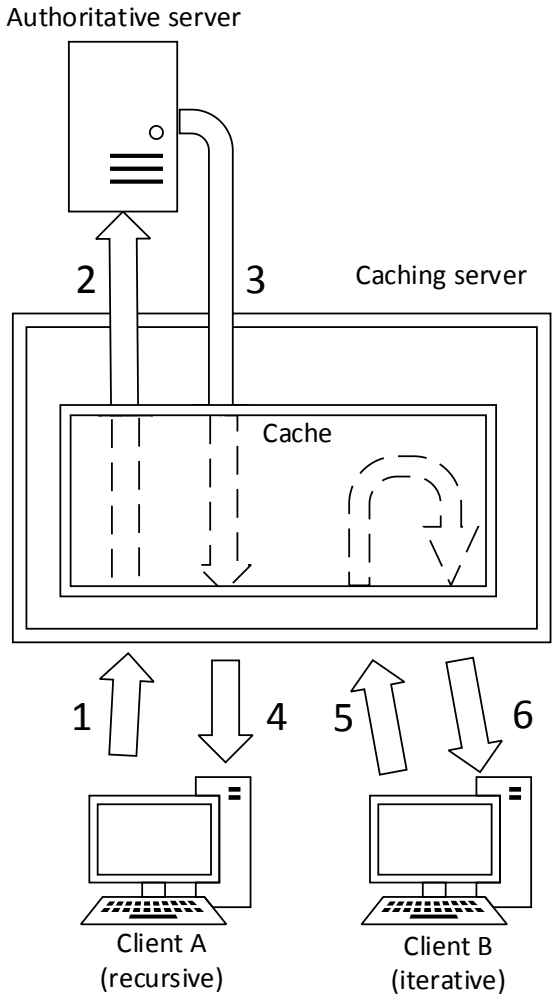


Figure 1. Architecture of communication between clients and server.

Table 1. Brief summary of communication principle.

	Bit 1	**Bit 0**
Send	Do recursive resolution	Do nothing
Receive-reading the TTL value of SOA	$TTL_{current} < TTL_{initial}$	$TTL_{current} \approx TTL_{initial}$
Receive-query time measurement	Time<Threshold	Time>Threshold
Receive-iterative request response code	NXDOMAIN response	NOERROR response

5 Address Space

The communicating entities must be aware of common address space before communication occurs. Uses of registered domain names and typos have to be avoided, since we expect *NXDOIMAN* responses. Each bit of data is addressed separately, so address space required for transmitting message is relatively large. Therefore queried domain names are generated

automatically by the program. To correctly interpret stored message both programs have to use the same address space with domain names in the same order. We decided to generate sequence of lowercase ASCII letters and digits (a – 999999 etc.). With maximal length of 63 characters, address space may be considered inexhaustible for our intention.

Example of an address format is shown below.

```
<sequence generator>.<unique identifier of the message>.<domain
without wildcard record>

1. bit:     a.9c52873f68574198a21691e178330a0a.sk
2. bit:     b.9c52873f68574198a21691e178330a0a.sk

or

1. bit:     a.192-168-1-5-msg20.subdomain.some-domain.sk
2. bit:     b.192-168-1-5-msg20.subdomain.some-domain.sk
```

6 Testing

The possibility of using these methods should be verified before the communication occurs. Testing program evaluates server (or server list) based on the received responses. Deciding on the appropriateness of using aforementioned methods is carried out based on the following assumptions:

- Sending is a failure when contacted server does not use the cache to store the responses. In this case, the message cannot be read by any of the approaches. Sending is also unsuccessful, if the server is unavailable or refuses to answer (response code *REFUSED* or *SERVFAIL*)
- The method of comparing values of the TLL of SOA record cannot be used for servers that perform NXDOMAIN redirection. Such response does not contain an SOA record.
- Receiving messages by iterative querying cannot be used for servers that does not accept this type of queries (response code *REFUSED*)
- Detection of record presence by measuring the query time when measured times for the present and non-present records are similar causes high error rate. In this case different address space may improve results.

Result of testing is showed in the Table 2. During testing computer was connected to a wired connection. The method suitability test was done successively on 158 public servers.

Test message for error rate measurement contained 3000 bytes of ASCII encoded text. Message was sent to public name server with lowest delay (approx. 25 ms). Transmission was done in 20 parallel threads. The effective average transfer speed during sending went up to 70 B/s and receiving speed went up to 40 B/s. We are aware of current implementational imperfection. Transfer speed is expected to rise after some improvements.

Table 2. Test result of proposed methods.

	Number of suitable servers	**Error rate**
Receive-reading the TTL value of SOA	82	0 %
Receive-query time measurement	88	<0.05 %*
Receive-iterative request response code	46	0 %

** Such error rate was achieved when fast caching server was used to resolve domain names with slower authoritative DNS server (10 times higher query time).*

7 Conclusions

We have verified that DNS cache can be used as a communication medium. The main disadvantage of using DNS cache channel is slow data transfer rate. To read a single bit of information program sends approximately 109 bytes and receives 166 bytes (actual frame size depends on the length of the domain name and the entry in the caching server). DNS cache is only temporary memory medium, since negative records are valid only for finite time.

Another disadvantage is the reliability of transmission. Server is not required to keep records in the cache until they expire. Early deletion would cause misinterpretation of the message.

The amount of transmitted data has to be controlled, since sudden increase in volume of DNS traffic could attract attention and reveal the channel. Server receiving a large number of DNS queries may interpret this as a form of attack. Some DNS servers use a variety of practices that limit the effectiveness of similar attacks[3]. DNS cache covert channel is not intended for transmission of large files. However, publishing small amounts of data (IP addresses) can hardly be detected.

Acknowledgement: This work was supported by Slovak Science Grant Agency VEGA, projects No. 1/0722/12.

References

[1] Andrews, M.: *Negative Caching of DNS Queries (DNS NCACHE)*. RFC 2308. [Online; accessed December 10, 2012]. Available at: http://www.ietf.org/rfc/rfc2308.txt.

[2] Bernát, D.: DNS as a Memory and Communication Medium. In: *SOFSEM 2008: Theory and Practice of Computer Science.* 34th Conference on Current Trends in Theory and Practice of Computer Science Nový Smokovec, Slovakia, January 19-25, 2008, (2008), pp. 560–571.

[3] *DNS Covert Channels and Bouncing Techniques*. Phrack Inc. [Online; accessed December 10, 2012]. Available at: http://archives.neohapsis.com/archives/fulldisclosure/2005-07/att-0472/p63_dns_worm_covert_channel.txt.

[4] Elz, R., Bush, R.: *Clarifications to the DNS Specification*. RFC 2181. (1997). [Online; accessed December 10, 2012]. Available at: http://tools.ietf.org/html/rfc2181.

[5] ICANN Security and Stability Advisory Committee. SAC 032 - *Preliminary Report on DNS Response Modification*. [Online; accessed December 10, 2012]. Available at: http://www.icann.org/en/groups/ssac/documents/sac-032-en.pdf.

[6] Kaminsky, D.: *Attacking Distributed Systems (The DNS Case Study)*. (2004). [Online; accessed December 10, 2012]. Available at: http://www.slideshare.net/dakami/bh-eu-05kaminsky-5939200.

[7] Mockapetris, P.: *Domain Names – Concepts and Facilities*. RFC 1034. [Online; accessed December 10, 2012]. Available at: http://www.ietf.org/rfc/rfc1034.txt.

[3] https://developers.google.com/speed/public-dns/docs/security

Security Modules for Measuring Tool KaTaLyzer

Tomáš HALAGAN*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
tomas.halagan@gmail.com

Abstract. Privacy and safety in today's world represents an important aspect of the daily activities in the field of all aspects of the human life. This is not so different in the field of informatics and information technologies. The aim of this paper is to investigate various Internet attacks and their derivations. The emphasis is placed on the selected DoS network attacks. Based on this information, the security modules are proposed for measuring tool KaTaLyzer, which is primarily used to measure and analyze network traffic. These modules provide important information for network administrators for ongoing active network attacks, thus are effective in preventing this attacks. This paper also contains detailed description of the proposed methods, data structures, implemented algorithms and functions of security modules. Functional security modules are subjected to prolonged and thorough testing with a lot of results through tools to simulate DoS attacks.

1 Introduction

Computer networks are a known phenomenon in the world of virtual communication. This part of information technologies, hand in hand with network attacks, is growing rapidly. However, the goal is not to create a defending tool for securing client's PCs, but to investigate all optimal ways to propose detection mechanism for quick reporting to network administrators. Currently, the basic protection against Internet attacks is implemented in different parts and levels of operational system, such as applications, the security protocols, firewalls , routers and intrusion detection systems. However, we cannot rely on certain security features. First of all, detection is one of the invaluable help as soon as possible to respond to emerging threats caused by Internet attacks. Time is important, the later we reveal the current Internet attacks, the more unpleasant consequences they will have on our system and this comes hand in hand with the prolonged time to restore the system back to normal. Extensiveness of the problem allows to deal with whole Internet attacks, some specific parts of the attacks, in this case the flooding attacks. In the following chapter two flood attacks are described, that security modules will detect.

* Master study programme in field: Computer Engineering
Supervisor: Dr. Tomáš Kováčik, Institute of Computer Systems and Networks, Faculty of Informatics and Information Technologies STU in Bratislava

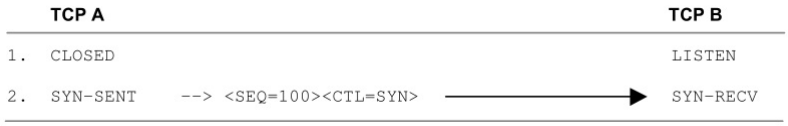


Figure 1. Half-open TCP connection.

2 DoS Attacks

2.1 SYN Flood

The SYN Flood attack [4] was one of the first DoS attacks to cause mass disruption among public servers. The attack exploits a rather obvious vulnerability in the connection setup stage of the TCP/IP protocol. Unfortunately during the design of TCP/IP the Internet was considered to be a "friendly" place. When a SYN Flood is initiated the following sequence occurs. The attacker changes the source address of the SYN packet sent to TCP B. TCP B will then try to send a SYN/ACK packet to this spoofed address. Normally if the spoofed address were an actual system, it would respond to TCP B with a RST (reset) packet, since it did not initiate the connection. However the attacker chooses to use the spoofed address of an unreachable system. Thus TCP B never receives a RST packet, and the potential or half-open connection in the SYN-RECV state is placed in a connection queue [3] as shown in Figure 1. This potential connection will only be flushed after a connection establishment timer expires. This timer can vary from 75 seconds to as long as 23 minutes on some systems. Because the connection queue is small, attackers many only have to send a few spoofed SYN packets every 10 seconds to completely disable a port. The victim system will never be able to clear the backlog of half open connections before receiving a new spoofed SYN packet. When multiple SYN flood packets are directed at a specific port on the victim machine, the service running on this port becomes starved of its resources.

2.2 UDP Flood

When a connection is established between two UDP services, each of which produces output, these two services can produce a very high number of packets that can lead to a denial of service on the machine (s) where the services are offered [5]. Anyone with network connectivity can launch an attack; no account access is needed. For example, by connecting a host's chargen service to the echo service on the same or another machine, all affected machines may be effectively taken out of service because of the excessively high number of packets produced. In addition, if two or more hosts are so connected, the intervening network may also become congested and deny service to all hosts whose traffic traverses that network.

3 Proposal

I choose this solution for various reasons:

- This solution uses a modified variant space-efficient probabilistic Bloom filter data structure. That structure in recent years more starts utilize the detection products for the network traffic as well as products for detection in other areas.

- This solution meets the aspect of efficiency and timeliness of similar products today

- This solution offer high usability and possibility to extend in the development environment.

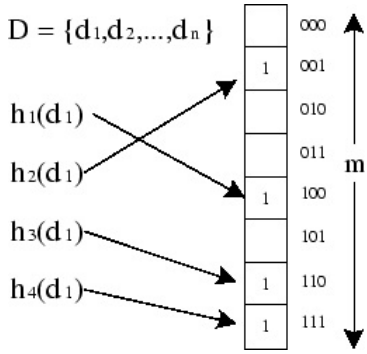
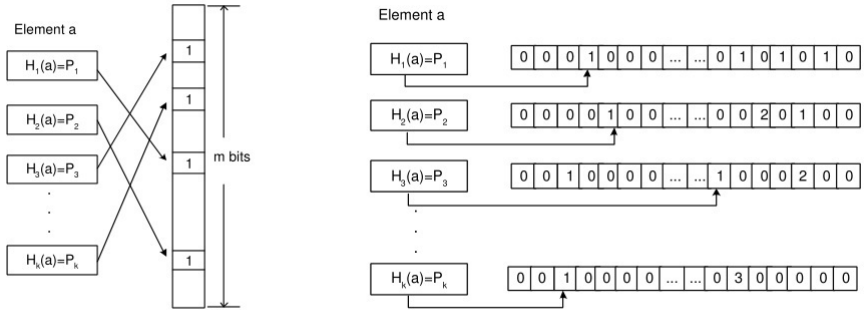


Figure 2. Bloom Filter.



(a) Original Bloom filter (b) Counting Bloom filter

Figure 3. Counting Bloom Filter.

3.1 Bloom Filter

Bloom Filter is a space-efficient probabilistic data structure that is used for record information. It was originally used to reduce the disk access time to different files and other applications, such as spell checkers. Recently, it has been adapted for use by some methods for defending against DDoS attacks. A Bloom filter is composed of a vector v of m bits, initially all set to 0. We have k independent hash functions, $h_1, h_2, \ldots, h_k$, each with a range $0, 1, \ldots, m-1$. The vector v can show the existence of an element from some address space D. Given an element $d \in D$, the bits at positions $h_1(d), h_2(d), \ldots, h_k(d)$ in v are set to 1 [2] as shown in Figure 2.

3.2 Counting Bloom Filter and Edited Solution

A variant of the original Bloom filter, called counting Bloom filter, uses a table of counters to replace the n bits, as shown in figure below. The table is composed of k rows with n counters in each row. The rows are independent of each other and each row corresponds to one hash function $h_i, 1 \leq i \leq k$. The n counters in each row correspond to addresses from 0 to $n-1$. All counters are initialized to 0. Each counter represents how many times the corresponding location has been hit. When a key a (such as an IP address) is inserted or deleted, the value of the corresponding counter in each row is increased or decreased by 1, according to $h_i(a)$ for all k rows as shown in Figure 3. If an IP address b is already stored in the modified bloom filter, the counters at locations $h_i(b)$, $1 \leq i \leq k$, in the table should all be nonzero.

Solution used for securing modules is, in fact, more simply than Counting Bloom Filter. After careful analysis, I came to the fact that the existing data structure can be for the purpose of designing and implementing security modules even easier. It consists only from one table with one counter, that will record results of all independent hash functions and according to the situation will be incremented or decremented. This simplification can be realized because the table with the values in a periodically time interval will be initialize to initial values, in this case, this interval is set to one minute. This resetting tables is used to prevent uncontrolled growth tables. For proper functionality of a quick solution based on the modified Counting Bloom filter is necessary to determine an effective independent hash functions that are described in the following chapter.

3.3 Independent Hash Functions

Having a good set of independent hash functions is essential for good hash table performance. Ideally, each hash function in the Bloom filter should hash the keys to the table uniformly. Moreover, the k hash functions are independent. In practice, it is not easy to design a good hash function that distributes the keys uniformly and yet has low computational cost. Moreover, the distribution of input keys affects the distribution of the counter usage. In this paper, we focus on the design of independent hash functions that have low probability of collision. We use the 32-bit IP address IP as the key of the hash functions. The hash functions are defined as follows: $h_i(IP) = (IP + \ IP \ mod \ pi) \ mod \ n$, $1 \leq i \leq k$, where mod denotes the modulus operation, n is the row length of the hash table, and pi is a prime number less than n. In comparative study performed by Mr Chen and Mr. Yeung in [1], n is set to 1024, k to 4.

3.4 Data-Flow Analysis

For analysis data-flow we used data flow diagrams, known as DFD, that helps us to understand the dynamic behavior of a program by examining the static code. The proposed security modules contains the following modules:

- Security module for detecting TCP SYN flood attacks – administrator of measuring tool KaTaLyzer starts measuring tool to measure and analysis network traffic. During runtime program in the specified time interval, in this case is default time interval equal to 1 minute, starts the process measuring network traffic. With this process running also analysis TCP connections, that is important for the detection performed by the security module. In data structure modified Counting Bloom Filter, that will be described in relevant same name chapter, are stored relevant informations obtained from analysis network traffic. Threshold value of maximum count an embryonic state SYNSENT (incomplete connections) is set to 50, because in known fact the optimum network traffic has around 5–10 these embryonic state.

  After detecting the value maximum count an embryonic state reach the threshold value, the process, responsible for creating a detailed record of the attack and also notify the Administrator warning information, will be running. Data-Flow diagram might look like as shown in Figure 4.

- Security module for detecting UDP SYN flood attacks – similarly as security module TCP SYN flood. In case of security module UDP SYN flood is not necessary to analyze network traffic. As we known, UDP is a connectionless transport protocol, entailing that a connection does not need to be established between the source and destination hosts. The point of this detection is based on maximum number of received UDP packets during a certain period of time, in this case is default time interval equal to 1 minute. Threshold value of maximum number of received UDP packets is set to 1000, because in known fact the number of received UDP is around 300–400 in optimum network traffic. This value is monitored and if reach threshold value, the alert information will created and send to administrator of measuring tool.

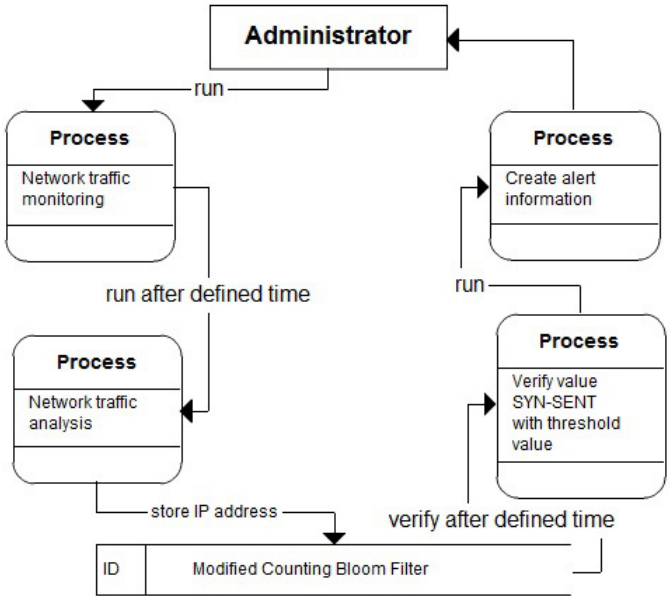


Figure 4. Data – Flow Diagram TCP security module

3.5 Proposed Algorithm

Proposal contains two hash tables, which are based on the edited Counting Bloom Filter. Relevant informations from network traffic about TCP handshakes, in case TCP securing module, is recording in that tables. First table, denoted as dstbloom, is used to record destination IP address. Second table, denoted as srcbloom, is used to record source IP address. In the first round of a TCP handshake are both IP addresses (destination and source) hashed using the k independent hash functions. Counters corresponding by the hash values are incremented. In the third round of a TCP handshake similarly again hashed using the k independent hash functions. Otherwise, counters corresponding by the hash values are decremented. If a TCP connection is correct establish, counters for both tables are first incremented and decremented, so that means no resulting changes.

3.6 Detecting Mechanism

For proper and fast detection current Internet attacks, if TCP flood attack, it is necessary to use a simple effective method that can immediately respond to an unwanted state. Algorithm to evaluate, whether it is an attack, consists in summing all the bits that contains modified data structure Counting Bloom Filter. In case of UDP flood attack is sufficient to monitor the number of received UDP packet.

4 Testing Security Modules

For testing security modules is desirable to have the amount of testing tools that are able to generate relevant Internet attacks. The proposed security modules are implemented under the operating system based on Unix – Linux. This fact reduces the number of useful tools, but well-known fact is the existence of a number of instruments, and even the environment, which is directly designed for this purpose. These environments include OS Backtrack 5, the tools contained in it are used for development and testing of security modules. Testing instruments contained in the environment as well as other tools are following:

– Hping is a command-line oriented TCP/IP packet assembler/analyzer. The interface is inspired to the ping unix command.To start generating TCP SYN flood is used the following command:

```
hping 192.168.2.132 -S -V -p 443
```

To start generating UDP flood with spoofing IP is used the following command:

```
hping 192.168.2.132 --udp --rand-source
```

– Letdown tool belongs to a widely used tool for creating TCP flood attacks. Examples of attacking might look like following command:

```
letdown -d 10.10.10.7 -s 192.168.1.132 -p 21
```

There are more number of simulation tools for creating web attacks. The instruments used to test security modules belongs ev1sync tool.

5 Conclusion

Created security modules for measuring tool KaTaLyzer are new in the field of measuring tools. For the first time measuring tool can detect Internet attack and about this fact can inform the network administrator. The proposed security modules as well as the measuring tool are designed and implemented according to strict rules of modular programming, which means a smooth extension of the measuring tool with the additional security modules in the future.

Acknowledgement: This work has been supported by ngnlab.eu project, 7FP project HBB-Next, Slovak National research grant 1/0676/12. It is a partial result of the Research and Development Operational Program for the projects Support of Center of Excellence for Smart Technologies, Systems and Services, ITMS 26240120005 and Research and Development Operational Program for the projects Support of Center of Excellence for Smart Technologies, Systems and Services II, ITMS 26240120029, co-funded by ERDF.

References

[1] Chen, W., Yeung, D.: Throttling Spoofed SYN Flooding Traffic at the Source. In: *Final Report*. College of Computer Nanjing University of Posts and Telecommunications, Nanjing 210003, Jiangsu, China, 2006, pp. 1–30.

[2] H. Wang, D.Z., Shin, K.G.: Detecting SYN Flooding Attacks. College of Computer Nanjing University of Posts and Telecommunications, EECS Department, The University of Michigan Ann Arbor, MI 48109-2122, 2002, pp. 1530 – 1539.

[3] Hutchinson, M.: Study of Denial of Service. In: *Final Report*. The Queen.s University of Belfast, Belfast, 2003, pp. 1–70.

[4] University, C.M.: Tcp syn flooding and ip spoofing attacks. Technical report, Software Engineering Institute Carnegie Mellon University, 1996, CERT/CC. 1996b.

[5] University, C.M.: UDP port denial-of-service attack. Technical report, Software Engineering Institute Carnegie Mellon University, 1996, CERT/CC. 1996c.

Advanced GLBP Load-Balancing (GLBP+)

Martin HREHA*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
martin.f.hreha@gmail.com

Abstract. Gateway Load Balancing Protocol (GLBP) supports several methods of load balancing traffic across GLBP group. However there is no method which provides some kind of feedback from AVFs used by AVG to properly load balance the traffic among available AVFs. Thus, under certain circumstances, a network performance decrease can occur due to the bottleneck existence. This paper presents solution of this problem – GLBP with advanced load-balancing called GLBP+. GLBP+ is capable to measure the utilization of AVFs within GLBP group and to inform the AVG about the measurement results. AVG then load balances the traffic according to utilizations of individual AVFs which prevents over-utilization of individual AVFs and consequent creation of bottleneck.

1 Introduction

Nowadays, people need and also want to communicate through computer networks. They use internet in everyday life and there are increasing demands on the availability and reliability of IT services. The basis for meeting these requirements is a proper function of communication networks.

In order to allow clients to communicate with the outside world (beyond the local network) the clients must send the traffic via default gateway which routes the traffic towards the destination. However, the device serving as default gateway is a single point of failure (SPOF). So when it fails all clients using it as their default gateway loose their connection to other networks. To avoid this, first hop redundancy protocols (FHRP) were introduced [2]. These protocols allow the group of several L3 devices to act as one default gateway. These devices share a common virtual IP address which is used by end devices like their default gateway address.

So in case that device responsible for packet forwarding in the particular FHRP group fails, another device from group takes over its function. This ensures that default gateway will be still available and end devices will not loose connection with other networks. But this would not be possible if the role of default gateway was assigned to a single physical device.

* Master degree study programme in field: Computer Engineering
Supervisor: Michal Olšovský, Institute of Computer Systems and Networks, Faculty of Informatics and Information Technologies STU in Bratislava

2 GLBP

One of the protocols from FHRP group is Gateway Load Balancing Protocol (GLBP). GLBP differs from other FHRP protocols in capability to forward traffic not only by one physical device, but by up to four physical devices in particular GLBP group [1]. These devices are known as active virtual forwarders (AVFs). The AVFs share a common virtual IP address, but they have unique MAC address assigned by Active Virtual Gateway (AVG) for that GLBP group.

AVG is elected from all devices in particular GLBP group and is responsible for group management. AVG is also responsible for load balancing the traffic (clients assignment) among individual AVFs. Load balancing is achieved in such way that the AVG replies to the ARP requests for virtual IP address with different virtual MAC addresses assigned to AVFs.

2.1 Load-balancing shortcoming

Because the traffic has to be load-balanced among AVFs, question of how best to distribute the load among individual AVFs arise.

In GLBP, there are three existing methods for traffic load-balancing [2, 3]:

1. Round robin – traffic is distributed evenly among all AVFs, client is assigned to the next AVF in the row.
2. Weighted – traffic is distributed based on weight value of individual AVFs, higher value results in more assigned clients.
3. Host dependent – each client always receive the same virtual MAC address in ARP reply for virtual IP.

Only weighted method, allows controlling the portion of hosts assigned to individual AVFs. Amount of assigned clients depends on actual weight value of individual AVFs. Number of assigned clients and the resulting traffic load handled by AVFs can be increased or decreased by changing the weight which can be done statically or dynamically [4].

However even this method can not eliminate the bottleneck existence. Let's assume a GLBP group with 3 AVFs with weight ratio of 4:2:4 (Figure 1). Each AVF has a group of assigned clients whose communication it must forward. Due to the smaller weight, the fewest number of clients is assigned to R2. But despite this, there is a possibility of forming a bottleneck in case that assigned clients produce heavy traffic. However as AVG has no feedback about the congestion, it continues in assigning other clients to R2.

To sum it up, the main disadvantages of GLBP is the absence of any feedback from AVFs which would provide AVG some details of AVFs utilization in particular GLBP group.

3 GLBP+

We have decided to eliminate this GLBP disadvantage by implementing feedback from AVFs based on their utilization measurement. Measurement results are then used by AVG for proper client load-balancing across GLBP group, so adequate number of clients would be assigned to every AVF. Thus, this feedback helps to avoid over-utilization of individual AVFs and consequent creation of bottleneck.

3.1 AVF utilization measurement

The best way how to measure AVF utilization is to measure bandwidth percentage utilization of selected interfaces (1).

$$\frac{\text{utilized interface bandwidth}}{\text{interface bandwidth}} \tag{1}$$

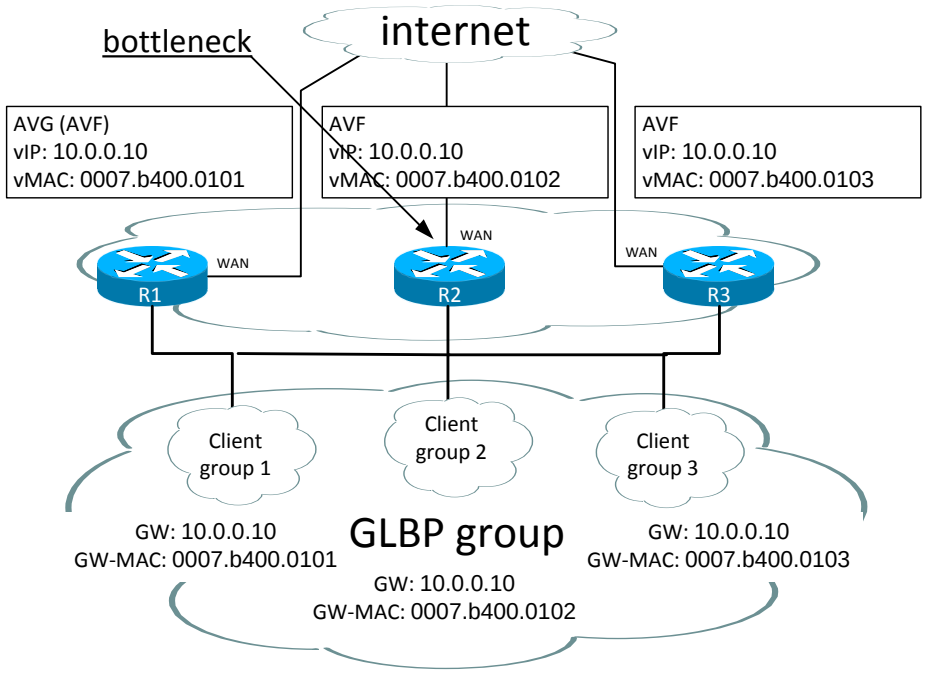


Figure 1. Illustration of GLBP group with bottleneck.

Utilization must be measured for each selected interface and the greatest measured value is used as a final AVF utilization.

Network congestion mostly occurs on WAN interfaces because as they have relatively small bandwidth compared to LAN interfaces. Therefore the utilization is measured on these interfaces by default. But when the network architecture is not designed properly the congestion can occur on LAN interface as well. Therefore there is a possibility to manually enable utilization measurement on all other network interfaces. There is also possibility to disable measurement on selected network interface

It is necessary to avoid peaks in utilization measurement. Peaks can cause that AVG would has distorted information about utilization of AVFs which can eventually cause wrong traffic load-balancing across GLBP group and even a bottleneck in an extreme case. Therefore the AVF utilization is measured as average value for a time interval dictated by AVG. This interval starts at low value and gradually increases to defined maximum. This measurement prevents inaccurate clients load-balancing among AVFs in early stages of forming a GLBP group.

The interval value is set in seconds and sent in hello submessage. Two bytes dedicated for its transfer allows setting the interval large enough. When AVF receives zero interval value it assumes that AVG do not supports GLBP+ and it stop measuring the utilization. However, when the non-zero interval value will be received, the measuring will be resumed immediately.

3.2 Sending the utilization value

It is necessary to keep backward compatibility between new GLBP+ and original GLBP. Therefore there was a demand for using the same format of GLBP messages.

Due to lack of documents describing GLBP messages, messages format was determined based on real GLBP traffic capture and analysis. Preserving of message formats was possible due to unused fields in GLBP submessages which are shaded in the following pictures (Figure 2 and Figure 3).

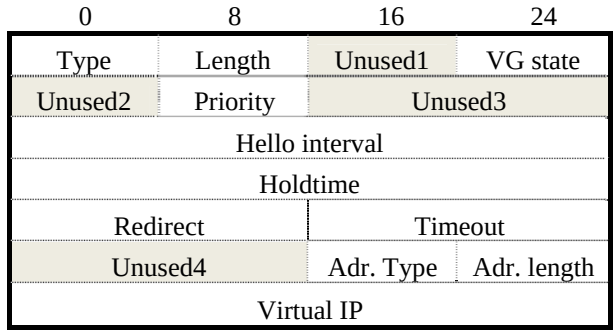


Figure 2. Hello submessage format.

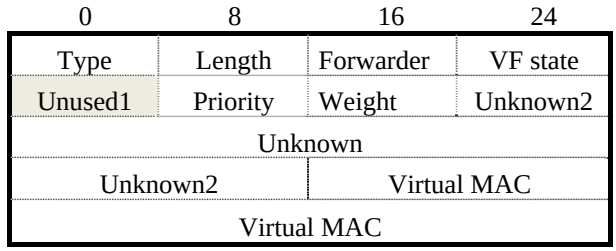


Figure 3. Request/response submessage format.

GLBP messages mainly consist of two types of submessages (Figure 2 and Figure 3):

- Hello submessage
- Request/response submessage

However request/response submessages are not suitable for sending AVF utilization value due to following reasons:

- Unlike hello submessage, request/response provides insufficient amount of unused space
- Every GLBP device has only one value indicating its utilization, but can send multiple request/response submessages. Therefore, it makes no sense to send AVF's utilization value in request/response submessages. In case AVF takes over role of another failed AVF this value can be then sent more than once which would be incorrect.

Therefore, the hello submessage is used for announcement AVF utilization.

As the main goal is to use GLBP+ in production environment, AVF utilization value should not take a lot of bytes in hello submessage. We have decided to use only one byte value, which means that the value can vary from 1 to 255 including and 0 means that utilization measurement is disabled on AVF.

3.3 Load balancing according to AVF utilization

Assignment of clients to particular AVFs is always in charge of AVG and this rule also applies to GLBP+.

Active AVG listens to every GLBP message from all routers in particular GLBP group. It checks if the message is received from AVF router. If so, it remembers the AVF utilization value which is used for further load-balancing.

During the load-balancig process new client is assigned to the AVF with the lowest AVF utilization. When two or more AVFs have the same AVF utilization value, the client is assigned to AVF with the longest interval from the last client assignment.

This load-balancing method must also meet the rule of Redirect timer which starts when a router from the GLBP group acting as AVF fails and another router from the same group takes over his AVF function. It says that no client can be assigned to replacement AVF after expiration of redirect timer [1]. Therefore, after the expiring of redirect timer, the ARP replies for virtual IP address from active AVG cannot contain virtual MAC address assigned to failed AVF.

3.4 Changes needed in current GLBP

GLBP+ is designed in order to eliminate any significant changes in GLBP. Message formats are kept as well as final state machines of AVG and AVF listeners. Their states and transitions between them remain the same. Only some new operations are added to certain states.
New components were added into original GLBP in order to measure AVF utilization and load-balance traffic.

This results in GLBP+ full backward compatibility with GLBP. Original GLBP simply ignores AVF utilization values used by GLBP+.

4 Expected Results

While using GLBP unbalanced utilization of individual AVF routers can occur. GLBP+ is designed to suppress this situation.

Let's assume a group with 3 AVFs (R1, R2 and R3). While router R2 has for some reason relatively high AVF utilization, other routers do not. This situation can consequently lead to the formation of bottleneck on R2. However this cannot happen in case of GLBP+ which is capable to identify AVF utilization and assigns clients according it. According to GLBP+ design the expected result is balanced utilization among AVFs in GLBP group (Figure 4).

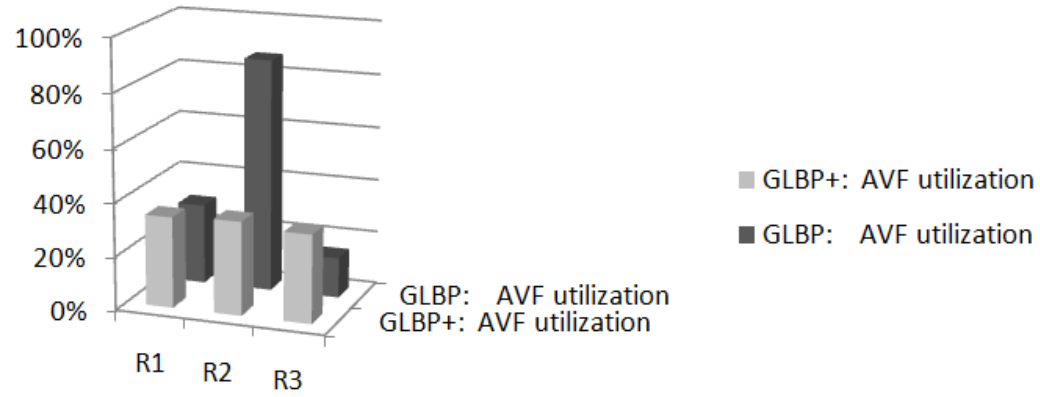


Figure 4. Comparison of AVFs utilization (GLBP and GLBP+).

5 Conclusion

The result of this work is the extension of GLBP called GLBP+. GLBP+ brings the missing feedback from AVFs. It helps the AVG to better load-balance traffic across GLBP group and also improves the congestion avoidance. Better load-balancing also allows faster traffic forwarding from end clients on AVF routers, as clients are not assigned to overloaded AVFs which could

bring additional delay into communication. Proposed GLBP+ is fully compatible with existing GLBP which allows the usage of GLBP+ in production environment.

Acknowledgement: The support by Slovak Science Grant Agency (VEGA 1/0676/12 "Network architectures for multimedia services delivery with QoS guarantee") is gratefully acknowledged.

References

[1] Cisco IOS Release 12.2(15)T: GLBP – Gateway Load Balancing Protocol. [Online; accessed February 20th, 2013]. Available at:
http://www.cisco.com/en/US/docs/ios/12_2t/12_2t15/feature/guide/ft_glbp.pdf

[2] Hucaby, D.: *CCNP SWITCH 642-813 Official Certification Guide*, Indianapolis Cisco Press, vol. 1, pp. 280-288, (2010).

[3] Cisco GLB Load Balancing Options [Online; accessed February 20th, 2013] Available at: http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6554/ps6600/product_data_sheet0900aecd803a546c.pdf

[4] Gateway Load Balancing Protocol [Online; accessed February 20th, 2013] Available at: http://www.cisco.com/warp/public/cc/pd/iosw/prodlit/glbpd_ds.pdf

Visualization of Digital Systems Models

Martin HYBEN, Tomáš JANČIGA, Martin KARDOŠ, Ľubomír MARON, Zsolt SÜLL*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
hdlteam@googlegroups.com

Abstract. The digital systems models complexity requires often representation of their structure in graphical form. Also the simulation results of the models are often very extensive and for their effective browsing the proper representation is needed. We have proposed the methods for visualization of the digital systems models structure as well as their simulation outputs. The methods were designed to support the three most popular hardware description languages (HDLs) – VHDL, Verilog and SystemC. For each language the algorithms were developed for transformation of models structure and simulation results into the common output forms that can be stored in common file formats for all the languages. To verify the solution all the methods were integrated into one system allowing to browse connections between modules on each level of model hierarchy. The simulation results can be displayed in the traditional waveform style, however the values of ports and signals at a particular time can be displayed directly in the visualized structure as well.

1 Introduction

HDL (Hardware Description Language) model visualization and simulation is very important. Using visualization and simulation of the models we can test their functionality and accuracy without building a hardware prototype. At present, there are many solutions for HDL models visualization and simulation, but most of them are either restricted to one HDL, or represent expensive commercial systems. Therefore our goal was to design and implement an interactive system for visualization of HDL models and simulation results that would support the most commonly used HDLs, namely VHDL (VHSIC HDL), Verilog, and SystemC, and would be easily extensible to include other languages.

2 Related work

The idea of HDL codevisualization or at least an attempt to clarify the code is not new. Together with the HDLsdevelopment the tools that helpedto visualize the individual parts of the system during the design have been developed. Nowadays it comes to complex applications that support visualization in multiple languages. An example of such complex solution is RTLVisionPRO [1] that is rendering support for VHDL, Verilog, and SystemC, similarly to the solution presented in

* Master degree study programme in field: Computer Engineering
Supervisor: Dr. Katarína Jelemenská, Institute of Computer Systems and Networks, Faculty of Informatics and Information Technologies STU in Bratislava

this paper. The application supports incremental compilation and also re-compilation ofselected parts. A similar solutionrepresents Mentor GraphicsDesignerHDL [2]designed for VHDL, Verilog, and System Verilog. HDL Designer is a complex system containing all the components necessary for digital systems design, visualization and simulation. Both the solutions are developed as a commercial software, however, visualization is a discussed topic in academic domain as well. For example at the University of Southern Methodist, Dallas PLFire [3]was developed to visualize the behaviour of a phased logic circuit. The topic has been studied at the Faculty of Informatics and Information Technologies, Slovak University of Technology in Bratislavaas well. For example in [4] authors dealt with the visualization of digital systems described in VHDL. On the other hand, Verilog models visualization is supported in the tool discussed in [5] and SystemC models support is presented in [6]. As far as the authors' knowledge extends none of the academic solutions supports more than two HDLs and the commercial solutions are highly complex to be appropriate for learning process. Therefore we decided to develop a new system that would support the most commonly used HDLs and would be easily extensible to support new language.

3 The HDL Visualizer design

The proposedsystem architecture is given in Figure 1.Analysis of VHDL and Verilog is based on classes, generated by the parser generator, ANTLR [7] that is based on a languages input grammar. The resulting modules VHDL2XML and Verilog2XMLare transforming source files to an XML file. The transformation is based on analyzed information. In case of SystemC specification, the compiler builds the specified model based on the modified library, which results in an Executable model. This model extracts the data from SystemC specification to XML file. The XML file can be represented as a temporary storage for analyzed data. The visualization of the specified system and other operations profits from the file. The VCD file contains information of how signals are changed in time. A source of this information is an output from the respective external simulators (GHDL, Icarus Verilog, and gbd respectively).

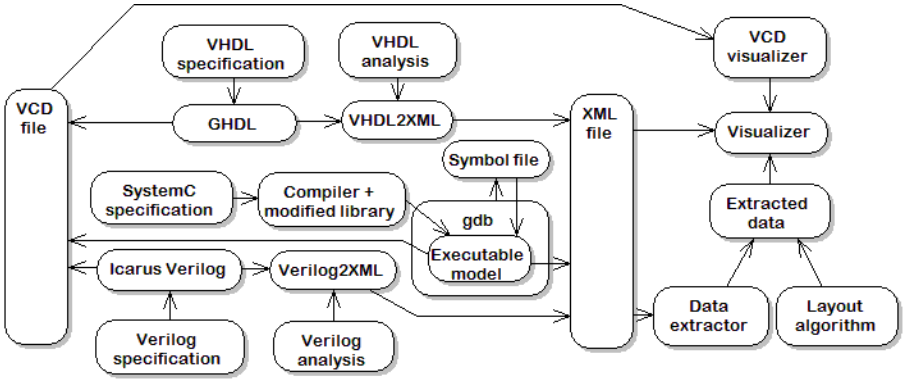


Figure 1. Architecture of the system.

The Data extractor is intended to read and extract data from generated XML file. It creates all objects and data structures for future work. In addition, the extractor takes care of attributes of previously created objects and structures. The layout algorithm works with the extracted data. It calculates the best effort location of each module and then inserts x and y coordinates to the internal memory.

The module Visualizer draws all the required objects to the GUI. The simulation results visualization is the main activity of the VCD visualizer that is displaying the results in two ways – the signal values are displayed in a waveform and in the generated scheme.

3.1 XHDL format, VCD format, and XHDL data extraction

To represent the information extracted from different HDLs the universal intermediate notation was introduced based on XML (eXtensibleMarkup Language). Although there are many specific XML schemes for source code transformation, for our purpose it was more convenient to design custom XML format that was called XHDL. It is simple, efficient and it stores information about the used descriptive *language*, *modules* (name, type, position, etc.), *ports* (name, orientation, data type, number of bits, etc.) and *signals* (list of connected ports, a set of interconnected nodes and their coordinates, etc.). The majority of XML data are extracted directly from an HDLcode, however, the information about position of graphical objects are added after the initial visualization. This allows the model re-visualization directly from the XHDL file without opening the source code files.

The HDL model simulation results, generated by different external simulators based on the specific language, are stored in a common VCD (Value Change Dump) format that is then used to visualize the simulation results.

VHDL and Verilog. The analysis of VHDL and Verilog files is based on classes generated by the ANTLR parser generator. The input grammars and parsers were inspired by [4] and [5]. For analysis purpose two modules, VHDL2XML and Verilog2XML, were developed that generate the XHDL data in several different steps. For VHDL the steps include:

- Selection of top-architectures from the description, and their conversion to modules.
- Definition of each module's ports.
- Definition of each top-architecture's signals.
- Recursive search and conversion of components to modules, creating their ports and signals.
- XML file creation storing the created elements and language information.

The relevant steps for Verilog include:

- Selection of top-modules and the list of top-modules creation.
- Searching of all module's instances in top-modules and creation of the list of instances.
- Scrolling through the list of top-modules and instances.
- List of each top-module's signals creation.
- List of each instance's signals creation.
- List of each top-module's ports creation.
- List of each instance's ports.
- Adding the module to the list of modules creation.
- XML filegeneration saving all lists (modules, ports, signals) and the language information.

SystemC. SystemC [8] differs from other languages for hardware and system description and specification. It allows designers to model systems and embedded software resources on different abstraction levels. In the case of SystemC, the model structure is assembled when the program is loaded into the memory and all objects are created. Consequently it is not so simple to extract the SystemCmodel structure using text parser. This approach would require the C++ compiler modification as in [9]. Therefore in this system a different approach was proposed based on SystemC library modification. Since the SystemC library interface that is used by user remained unchanged the standard library can be replaced by our enhanced solution without any changes in the models source codes. When the model is compiled and linked together with the modified library the executable file is created.

The model structur eextraction has been implemented in the function simcontext_to_xml(). This function is called using standard SystemC function sc_start(), which executes simulation of

the model, so this function is called after the whole model's structure is loaded and assembled. Functionsimcontext_to_xml() has one parameter, which is object of the sc_simcontext class. This object contains pointers to all objects of the model. There are objects grouped into modules ports and signals in the function sc_to_xml.

When extracting the objects attributes the SystemC and C++ names inconsistency had to be addressed. The C++ names can't be extracted in the form of string from the source code, because C++ doesn't provide this feature. When the SystemC names are used instead of those from C++, the names can be automatically generated and user will not be able to associate them with the real object names from the model source code. It was therefore necessary to analyse the source code to extract the C++ names. To make this possible, all source files should be converted into the strictly defined format. For this purpose we used the gdb debugger for C++ language that allows us to stop a program execution, and write the symbol file, whose format is unambiguous.

The model's executable file must be executed by the gdb. When the program calls our function simcontext_to_xml(), it is stopped by interrupt number 3. This interrupt represents a breakpoint for the gdb. Now the symbol file is saved. The execution of program continues in the simcontext_to_xml() function, and the symbol file is opened. The C++ names of objects are assigned from the symbol file, when these objects are classified. The data about classified objects are finally written into the XHDL file. There are also written their attributes and connection information.

3.2 Locations of modules, ports and signals

For the modules, ports and signals layout the algorithm has been developed that concentrates on best arrangement of graphical elements and thus to simplify visualization. The layout parameters are stored in the XHDL format for future re-visualization.

Location of modules takes into account proportions of graphical interface window, length and number of line breaks or crossed signals. It is essential to create a list of interconnections between modules, i.e. list of module pairs connected by signal. This creates a grid with calculated or pre-entered dimensions. The first and last column of the grid is used for module ports of hierarchically higher level. First, modules are arranged in columns, which can be achieved by placing the module with output port as close as possible to the left of other module with input port (mentioned modules comes from prepared list of interconnections between two modules). Later, modules are re-ordered in each column so that the length of the line between interconnected modules is as small as possible to decrease the risk of intersection. Module height is derived from the number of ports. It is also necessary to estimate the size of the gaps between modules with the proposed formula:

```
Gap_size = 100          //minimum size in pixels
if (interconnection_count> (2 x modules_count))
gap_size = gap_size + ((interconnection_count / modules_count) x 40)
```

Ports are arranged in the order as listed in the XHDL file, and input ports are located on the left-hand side of the module and output or bidirectional ports on the right-hand side.

Regarding the signal layout, it is necessary to create a grid whose elements are gap intersections between modules. Also the list of interconnections between modules is used. At the beginning, there are recoded direct connections of two modules, or links with at most two breaks. This means that we store the positions of line breaks which form interconnection. For each interconnection (part of the signal between two modules) is given priority, taking into account the length of the line, but the main focus is on other interconnections forming a signal so that the signal is branched. For a better look of visualization, links are removed over the first row of modules and underneath the last one. Links with the highest priority will remain, all others are deleted. Substituting real coordinates we obtain signal routes, even if they overlap. To avoid overlapping of signals, each horizontal and vertical space between modules is divided into as

many parts as there are lines in this area. Lines are distributed into the area, and always tested whether the distribution was successful and whether it can be improved, as can be seen in Figure 2. The result is a set of points which are the start and end nodes of signal and also coordinates of all line breaks.

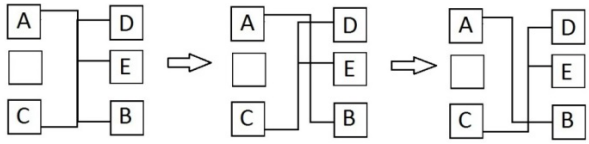


Figure 2. The distribution of linesand testing for improvement.

3.3 Simulation and simulation's results visualization

An important part of VHDL or Verilog specification is a TestBench. It can be specified both, implicitly or explicitly by the designer. An explicit specification of the TestBench is supported by an interactive window. This offers manual definition and automatic generation of the signal's value assignment in the defined time. For the purposes of simulation the external simulator GHDL [10] (VHDL) and Icarus Verilog [11] (Verilog)are used. The syntax of a model is checked and the model is compiled before the actual simulation. The simulation results are saved into the VCD file.

To generate the simulation results for SystemC models the VCD file is created and the simulation results are written into it. The modified SystemC library allows tracing of all objects classified as port or signal. The C++ names are already extracted when the simulation is started and they can be used for VCD file initialization. The standard SystemC initialization of VCD files had to be modified, because it writes all objects from different hierarchical levels into one global level and information about hierarchy of the model is dropped.

There are two possibilities for simulation results visualization. First, the external program GTKWave [12] can be used to display the signals values in traditional waveforms. Second, the simulation results can be displayed directly in the generated model structure. For this purposes the VCD2XML module, inspired by [5], was developed. The analysis of VCD files is based on the defined grammar and the analysed data are stored in the internal memory.

There are differences between VCD files generated by the GHDL and Icarus Verilog. These include headers, variable types and top level signals definitions. Different headers and variable types are not a problem. Unfortunately, the top level Verilog module is considered as an instance. On the other hand, the top level VHDL entity is described by architecture and is not considered as an instance. This difference results in a problem, because on the top of the list of signals definition the scope "$scope module module_name $end" is not created. The solution was an initial scope-free signals analysing function.

After the analysis of the VCD file using VCD2XML module, the internal memory stores time points, when values of signals have changed. The user can then choose a time, when the values are requested. Values are visualized next to ports of modules both by number and colour. The comparison of two ways of simulation results visualization of the "sum1" module in the time 6 seconds is given in Figure 3.

4 Conclusion

The aim of the work, presented in this paper, was to develop a system that will provide an intuitive interface and enable in the same way to visualize the structure and simulation results of models described in different languages, specifically VHDL, Verilog, and SystemC. An important requirement for the final system was the possibility to extend the support to other languages in an easy and straightforward manner. This was made possible by the design of an appropriate format

of data extracted from the model. For each language it was therefore necessary to design and implement an appropriate method for extracting the information about the model structure.

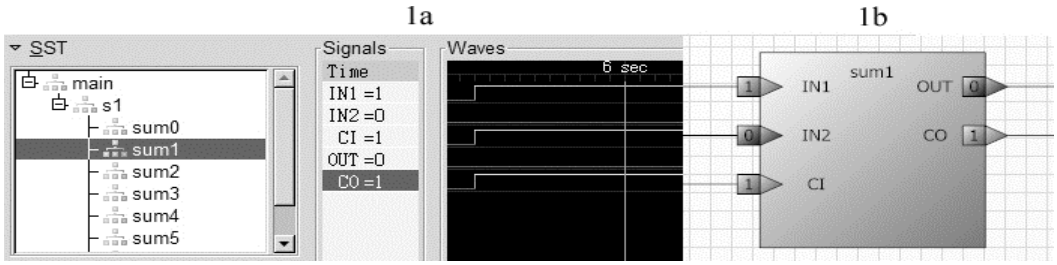


Figure 3. The comparison of ways of simulation's visualization.

For the storage of simulation outputs a commonly used VCD file format was selected. This is why we can use an existing program GTKWave to visualize this information. However, an appropriate method for VCD file generation based on simulation of models described in various languages had to be proposed and implemented.

Acknowledgement: This work was partially supported by Slovak Science Grant Agency (VEGA 1/1008/12 "Optimization of low-power design of digital and mixed integrated systems").

References

[1] Concept Engineering GmbH:RTLVision Pro – Visualize, Debug and Integrate RTL Code [Online; accessed February 22, 2013]. Available at: http://www.concept.de/RTLvision.html

[2] Mentor Graphics: HDL Designer [Online; accessed February 22, 2013]. Available at: http://www.mentor.com/products/fpga/hdl_design/hdl_designer_series/

[3] Fazel, K.; Thorthon, A.M.: PLFire: A Visualization Tool for Asynchronous Phased Logic Designs [Online; accessed February 22, 2013]. Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.11.6026&rep=rep1&type=pdf

[4] Macko, D.,Jelemenská, K.: VHDL Structural Model Visualization. In *EUROCON 2011*: International Conference on Computer as a Tool, IEEE, (2011).

[5] Jelemenská, K., Čičák, P., Nosáľ, M.: Visualization of Verilog Digital Systems Models. In *Emerging Trends in Computing, Informatics, Systems Sciences, and Engineering, Lecture Notes in Electrical Engineering*, Vol. 151, Springer, (2013), pp. 805-818.

[6] Turoň, J., Jelemenská, K.: Contribution to graphical representation of SystemC structural model simulation. In: *FPGAworld '10 Proceedings of the 7th FPGAworld Conference*, ACM New York, (2010), pp. 42-48.

[7] Parr, T.: *The Definitive ANTLR Reference: Building Domain-Specific Languages*. The Pragmatic Bookshelf, (2007).

[8] IEEE Standard SystemC Language Reference Manual: IEEE Computer Society, (2006)

[9] Sýkora. J.: *Metody extrakce modelu z jazyka SystemC*. Diplomová práca,: ČVUT v Prahe, Fakulta elektrotechnická, (2009), 89 p., [Online; accessed February 22, 2013]. Available at: http://necago.ic.cz/prj/sc2vhdl/dip-20090512-rev3.pdf

[10] Gingold, T.: GHDL, [Online; accessed Feb. 22, 2013]. Available at: http://ghdl.free.fr/

[11] Icarus Verilog, [Online; accessed Feb. 22, 2013]. Available at: http://iverilog.icarus.com/

[12] GTKWave, [Online; accessed Feb. 22, 2013]. Available at: http://gtkwave.sourceforge.net/

Atmospheric Modelling via Flying Platform

František KUDLAČÁK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
frantisek.kudlacak@gmail.sk

Abstract. Atmospheric model is very important in many useful applications. It is particularly useful in air sports, such as paragliding and hangliding, furthermore, it can be useful for forecasting weather. Local atmospheric model is created for small area, and provide information about dynamicity, temperature and behavior of air masses in modeled area. This paper deals with the hardware design of flying platform for gathering information about atmosphere. There is described software implementation of flight control system and implementation of modeling system with forecast module.

1 Introduction

Atmospheric models are nowadays created for weather forecast, but their accuracy is in many cases insufficient because of many influencing factors. In air sports there is a need to predict precise information about air masses in a particular area. These predictions have to describe behaviour of clouds, temperature and humidity, and also the special characteristics like dry adiabatic curve, humid adiabatic curve and temperature gradient. Considering land specifics lead to creating models which describe complex atmospheric behaviour and allow us to predict or visualise thermal flows in specific area.

The second role of atmospheric models is visualising state of atmosphere in real time. This visualization can help pilots to make good decisions before flight.

There are four main types of models known:

1. Global actual state model – describes the actual state of atmosphere.
2. Global forecast model – derived from first type, predicts atmospheric situation in the future.
3. Local actual state model – created using the special balloons and stationary station; describes actual state of local atmosphere.
4. Local forecast model – derived from all other types, predicts precise atmospheric situation in a small area.

The proposed solution should implement the third and fourth type of atmospheric model. It has to contain a device for measuring the data and software for creating the model. In this paper, we describe complete co-design of this solution. The paper is divided in three main parts:

---

* Master degree study programme in field: Computer Engineering
Supervisor: Dr. Mária Pohronská, Institute of Computer Systems and Networks, Faculty of Informatics and Information Technologies STU in Bratislava

1. Hardware design.
2. Software design of the measuring device.
3. Software design of the program that creates the atmospheric model.

Prototype and testing in real conditions are part of this paper, too.

2 Related work

There are some common global forecast models related to global state models, which use information from local state models. Each of these models uses different algorithms for prediction and modelling. These global models are very demanding for computing resources. There are very few if any complete models for local forecasting, however, local forecasts are nowadays derived from global forecast with very bad precision for small areas. This is caused by resolution of fragments in the global model. The fragments in the global model represent parts of the surface and the atmosphere, average value of each attribute is computed for each part. The unique marks of surface (e.g., hills, valleys) are not included in the description of the fragment. So if the resolution of the global model is small, the local forecast will be inaccurate but the computing time will be short. Higher resolution provides better outcomes but the computing time is significantly increased.

2.1 Global state models

Global state models are created from three main sources:

1. Images of atmosphere taken by satellites. These images can be taken with many different wavelength filters.
2. Radars provide information about actual state of clouds. There are information about rain and storms, too.
3. Temperature and wind information is provided from local observing stations.

With these information there can be created global model.

2.2 Global forecast models

Global forecast models are computed from state models. The smaller areas are represented by their average values of attributes. There are two main forecast models globally used nowadays:

1. ALADIN – size of one piece is 9 km and there are 37 layers. Most common configuration is for forecasting for three days [1].
2. ECMWF – size of one fragment is 25 km and there are 90 layers. Forecast in most common uses is for ten or eleven days [2].

2.3 Local models

There are just few local models, mainly created for simulation of air masses in small area. These models do not match requirements for forecasting model or for air sports uses. There are some way how to get information about air masses in smaller areas. Meteorology balloons are good way how to acquired required information. But they are quite expensive and they do not cover the entire area, only some spots. Some information can be obtained by ground observing stations, but there are lot of stations needed to cover the entire area.

3 Solution proposal

3.1 Device specification

There are several functions which have to be implemented in the device:

1. Flying platform with high endurance (30 min).
2. Logging actual position in real time using the GPS module.
3. Logging altitude above sea level, combining data from GPS module and barometric sensor.
4. Measurement of air mass characteristics (e.g., temperature, humidity, vertical and horizontal air mass speed).
5. Autonomous movement in air using pre-programmed way.
6. Automatic actions in air such as taking off and landing procedures.
7. Possibility to switch between autonomous mode and remote mode during flight.
8. Connectivity with PC or mobile for transport of measured data.
9. Air stability, ability of hovering.
10. Emergency landing, when battery is too low.
11. Type of construction and used components have to be wind and humid proof. It has to be resistant against low temperature (about -10 °C).

3.2 Modelling software specification

There are required functions which have to be implemented in the software solution:

1. Connectivity with the flying platform.
2. Visualization of air masses.
3. Differentiate between desired thermal flows and between compensating drops.
4. Offer information about wind in different places.
5. Compute missing information using approximation techniques.
6. Based on day time provides forecast for 12 hours.

4 Hardware design of the flying platform

The design process was focused on fly time and on delay characteristic of system. Reliability of system is very important because one defect in construction can cause crash of the flying drone. There is need for combined control, autonomous and remote, so remote control has to be implemented. Final hardware configuration is shown in Figure 1.

Many different communication interfaces are used, plus we need four PWM pins to control the motors and six PWM pins to communicate with the RC module, which receives signal from the RC transmitter over six channels. Telemetry communication can send and receive data, so ground unit will be provided with online information (position, sensor values, battery voltage, mode) about the flying platform. The I2C interface is shared via all devices communicating by this interface. Each device can be addressed separately. There are two serial interfaces, first communicates with the GPS module, which was reused from the Vario project [3]. For controlling so many interfaces, the Arduino Mega 2560 is the suitable choice [4].

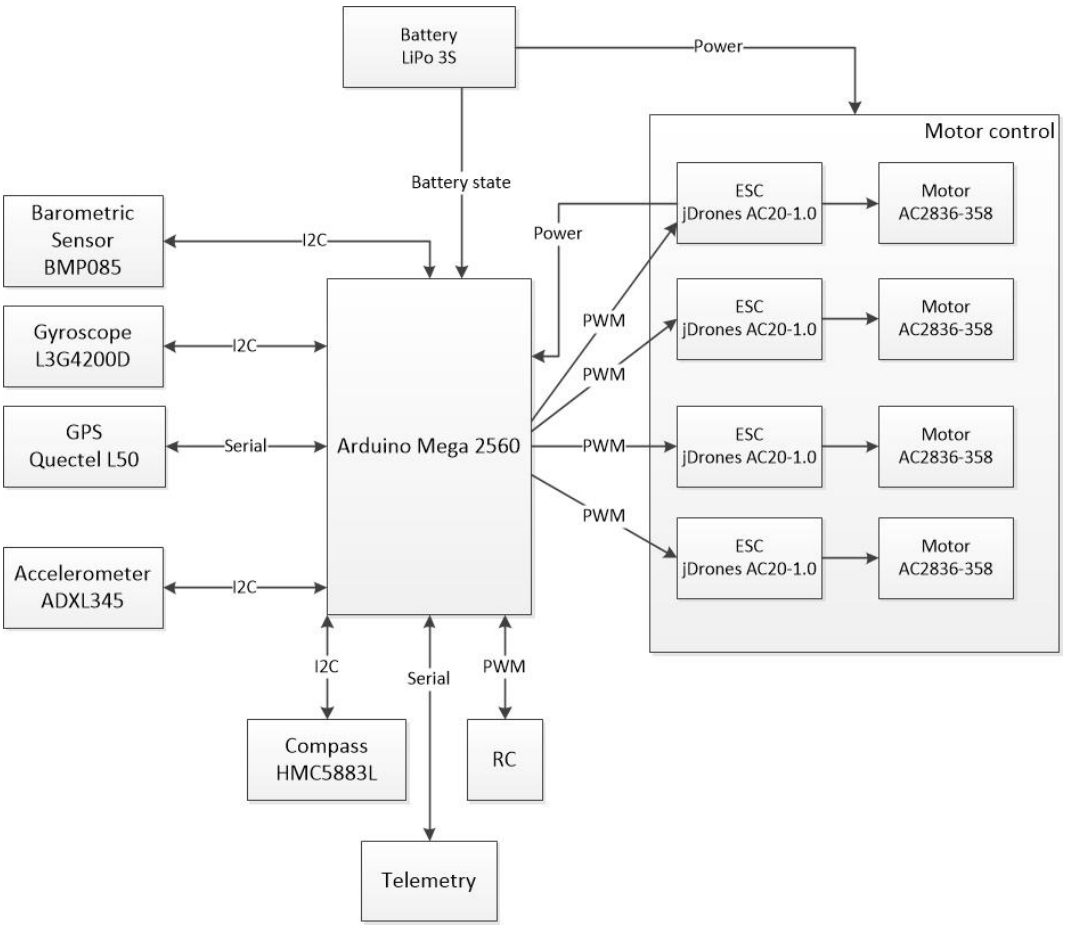


Figure 1. Hardware configuration and communication scheme.

5 Software design of the flying platform

The designed solution is based on modularity, so each part is implemented as a separate module and can be reused in other solutions. The program was written in the Arduino environment, using C++ programming language. We have developed libraries, which provide control of sensors. The essential function is the flight control, which stabilizes the flying platform and controls its movements. Interactions between modules are shown in Figure 2.

The main part of the program is the Operation module. Raw data are provided from the sensor module, this information about position and dynamic characteristic of platform have to be processed and scaled. These information are stored into data module. Second part of the Operation module is flight control, which controls stability and direction of flight. It provides the information to the motor control processing module, which then controls the motors. In case of remote control override, the motor control processing module takes information from the RC and Telemetry control module. All information are transferred via the RC, Telemetry control module.

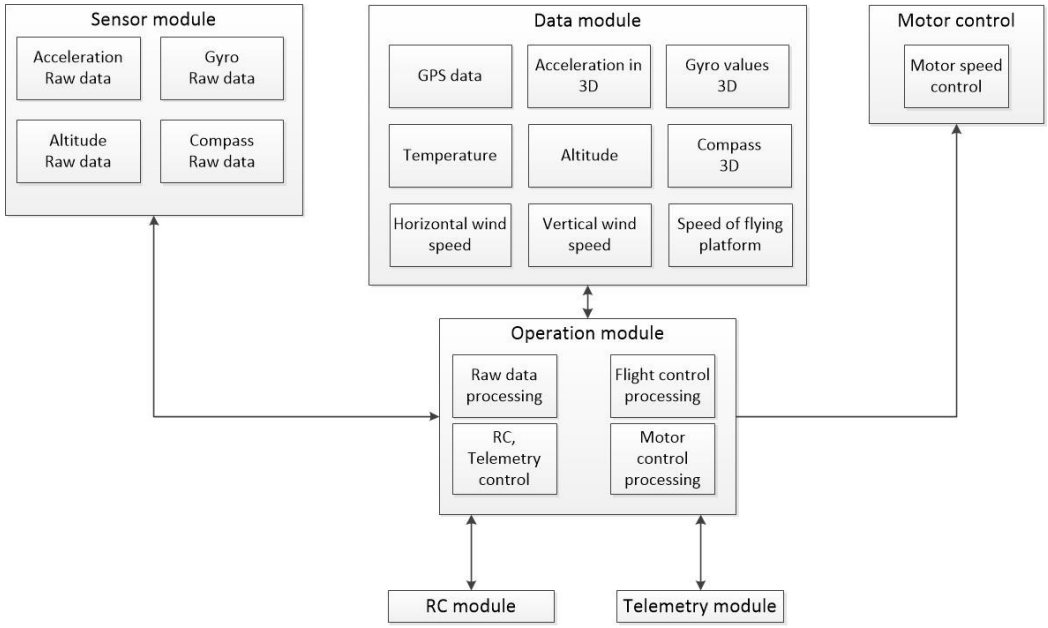


Figure 2. Interactions between software modules.

6 Design of the modelling software

The modelling software visualizes state of atmosphere in area and its dynamics characteristic. The structure of implementation is shown in Figure 3.

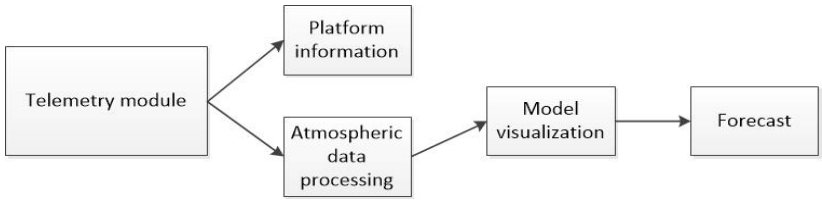


Figure 3. Visualization software structure.

The program has been implemented in C#. Telemetry module is hardware device connected to computer communicating via serial interface. The information from the sensors are received by the Telemetry module and further provided to the Platform Information module, which inform the user about state and position of the platform. The atmospheric model is based on the processed atmospheric data and position of the device in time. Platform observes the area in layers and the values between layers are interpolated. After creating the model, a forecast is computed.

7 Prototype and testing

Testing was performed on prototype (see Figure 4) in real environment. The flying platform should fly only in wind speed less than 15 m/s. Flying characteristics and stability are good. The flight time with small battery pack is around 14 minutes and with enhanced battery pack around 40 minutes.

Figure 4. The prototype.

8 Conclusions and further work

Developed local atmospheric model satisfies demands of air sport community. Special contribution in the field of hardware was creating the flying platform. This platform can be used in other applications because payload weight is around two kilograms. Contribution in field of software was development of the flight control system and development of the visualization and control system.

In the further work we can extend flying platform with another sensors (e.g., air quality sensor, light sensor, camera). Platform software can implement new functionalities like improved navigation capabilities and better autonomous behaviour. New forecast algorithms (e.g., neural network) would greatly improve functionality of forecast system.

Acknowledgement: This work was partially supported by the Grant No. 1/1105/11 of the Slovak VEGA Grant Agency.

References

[1] Vivoda, J.: *Model ALADIN*. [Online; accessed February 21, 2013]. Available at: http://www.shmu.sk/sk/?page=1016

[2] Minďas, J.: *Model ECMWF*. [Online; accessed February 21, 2013]. Available at: http://www.shmu.sk/sk/?page=1164

[3] Kudlačák, F.: Variometer with GPS Logger. *Bulletin of the ACM Slovakia,* (2012), Vol. 4, No. 2, pp. 47–50.

[4] Atmel: *8-bit Atmel Microcontroller with 64K/128K/256K Bytes In-System Programmable Flash. Datasheet.* [Online; accessed February 21, 2013]. Available at: http://www.atmel.com/Images/doc2549.pdf

Binary Decision Diagram Optimization Method Based on Multiplexer Reduction Methods

Marián MARUNIAK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
mar.maruniak@gmail.com

Abstract. In VLSI circuit synthesis, multiplexers are widely used as a basic element because of their ability to perform any Boolean function. Since multiplexers form a significant part of total circuit area, designers often focus on application of various optimizations. Multiplexer optimization techniques result in significant improvement in performance, area and power consumption of synthetized VLSI circuits. One of such approaches is the use of BDD as a structural representation of a multiplexer tree along with BDD optimization methods. This paper describes a BDD optimization algorithm combining a multiplexer reduction method with basic BDD reduction methods. Experimental results show that implemented algorithm reduces total amount of multiplexers in optimized multiplexer tree by 80% on an average compared to a non-optimized multiplexer tree.

A paper based in part on this paper was published in Int. Conference on System Science and Engineering (ICSSE 2013), IEEE, pp. 395-399.

* Bachelor degree study programme in field: Computer Engineering
Supervisor: Peter Pištek, Institute of Computer Systems and Networks, Faculty of Informatics and Information Technologies STU in Bratislava

Using the Methods of Artificial Intelligence in Network Detection Mechanisms

Igor SLOTÍK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
xslotiki@is.stuba.sk

Abstract. Various Network Intrusion Detection Systems are based on different packet features. Selection also affects the performance of the system and the type of detected attacks. In the fig. 1 is a schema that can map each of the proposed group of features into specific types of attacks. In this way it allows selection of relevant features serving for the successful attack detection. Because of effective feature storage and fast response time, we use cache between system and MySQL database. Consequently, our approach allows fast packet processing and it is on-line applicable.

1 Basics

Network security is currently rapidly developing area. The important components of network security are Network Intrusion Detection Systems (NIDS) that alert firewall when attack in network traffic is detected. NIDS are not able to effectively analyze all packet features, because of limitations such as huge computational time and challenging memory requirements. Large computational time can cause a late response to the attack. In addition, a large number of features selection does not necessarily lead to a more successful detections, but it can have a negative impact on system performance [1].

For these reasons, NIDS analyze only the relevant features of a specific attack or features relevant to a particular layer of the TCP/IP architecture. Our system allows on-line identification of suitable features for the detection of various types of attacks.

2 Feature Classification Schema

The proposed schema uses four main abstractions of the network security domain, such as network, host, connection and packet. Features are prominent characteristics of these abstractions. A network is here defined as any arrangement of interconnected hosts. Packets are the fundamental information carriage in the network, while a connection is referred as the act of bringing two hosts into contact

* Master study programme in field: Software Engineering
Supervisor: Assoc. Professor Ladislav Hudec, Institute of Applied Informatics, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 337–342.

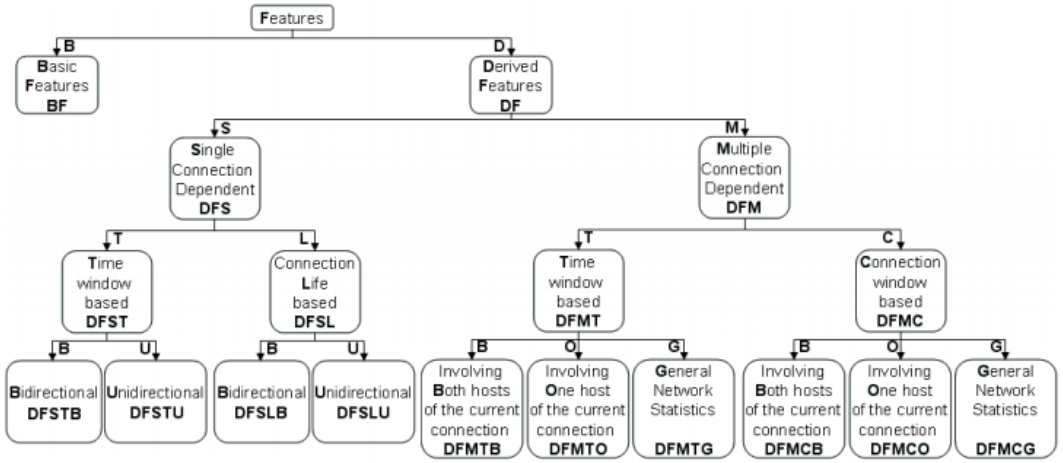


Figure 1. Feature Classification schema [1].

for bidirectional information exchange. The schema can be used to arbitrary protocols and layers of the TCP/IP architecture [1].

Figure 1 represents an overall view of the proposed feature classification schema. Features are prominent characteristics of network packets. The schema uses acronyms because of the large number of categories. For easy overview and schema orientation, the acronyms create a tree, where each edge carries a letter that adds to the end of the acronym [1].

Two main categories of features are Basic Features (BF) and Derived Features (DF). The BF category contains all features that can be extracted from a single packet field, while the DF category contains all features that require multiple-packet analysis over a period of time [1].

2.1 Basic Features (BF)

Any field of a packet is a possible candidate for this category, for example: source IP, destination IP, protocol, source Port (if applicable), destination Port (if applicable), flags (if applicable), ICMP Type (if applicable). This category also includes *timestamp* [1].

These features are vital for computing the DF category. For example, packet timestamp, is linearly increasing in time, its direct use in any kind of detection technique is not possible. However, it can be successfully used to compute DF features such as the duration of a certain connection [1].

2.2 Derived Features (DF)

Any features that require multiple-packet analysis over a period of time is contained in this category. The DF category can be further divided into two subcategories considering the number of connections that they belong to. The classification schema identifies single connection dependent features (DFS), and multiple connection dependent features (DFM). These subcategories are also applicable for connectionless protocols, since in this case a connection is represented as the number of packets exchanged between two hosts in the last x seconds (using the same two ports when appropriate) [1].

2.2.1 DF-Single connection dependent features (DFS)

This category contains all the features that can describe a single connection. DFS category is divided into these two sub-categories [1]:

– DFS-Time window based features (DFST): contains all the DFS features computed by using a time window interval.

– DFS-connection Life based features (DFSL): contains all DFS features computed with respect to connection's life period.

Each of the previously discussed subcategories (i.e., DFST and DFSL) can be further analyzed with respect to the direction of the information exchanged between two hosts of the connection. Consequently, the next subsection describes them in parallel. For easier notation let DFSx category stands for either DFST or DFSL categories [1].

DFST and DFSL features (DFSx)

Since there will always be two hosts involved in a single connection, each DFSx feature can be further describe unidirectional information exchange Unidirectional features (DFSxU) or bidirectional exchange Bidirectional features (DFSxB), where *x* can be replaced with either T or L for Time Window based and Connection Life based features, respectively [1].

The DFSxU features are computed with respect to either one of the two hosts that belong to the connection. For example, the "number of packets" feature will become "number of packets sent by source IP to destination IP", and "number of packets sent by destination IP to source IP". The DFSxB features are computed with respect to the contribution of both hosts at the same time [1].

2.2.2 DF-Multiple connection dependent features (DFM)

Symbols AAA stand for the set TCP, UDP, ICMP and symblos BBB stand for the set TCP, UDP. This notation is used in Table 1 and Table 2.

The DFM category is further separated into the following subcategories [1]:

– DFM-Time window based features (DFMT): includes all the DFM features that are computed with respect to the last time interval,

– DFM-Connection window based features (DFMC): includes all the DFM features that are computed considering the last *n* encountered connections.

Each of the previously discussed DFM subcategories can be further analyzed versus the connections that are taken into consideration when the features are computed. Consequently, the next subsection describes the DFMT and DFMC categories in parallel. For easier notation let DFMx stand for both DFMT and DFMC categories [1].

– DFMx-involving Both hosts of the current connection (DFMxB): this group of features is computed considering all the connections between HostX and HostY, where the two hosts are found in the currently sniffed. Subset of DFMxB features are in Table 1. packet [1].

– DFMx-involving One host of the current connection (DFMxO): this group of features is constructed in order to monitor the interaction between one host (HostY). and the whole network [1]. Subset of DFMxB features are in Table 2.

– DFMx-General network statistics (DFMxG): this group of features is used to provide statistical information about the state of the whole network [1].

Table 1. DFMxB features [1].

Number	Features
1	no. of created (or used) AAA connections between host HostX and HostY
2	no. of BBB connections created by HostX using any port (or PortV) to connect to any port (or PortW) on HostY
3	no. of packets (or bytes) sent by hostX to HostY
4	no. of TCP packets sent with FlagZ from HostX to HostY
5	no. of ICMP echo request packets sent by HostX to HostY
6	no. of ICMP destination unreachable packets sent by HostY to HostX
7	Average no. of AAA bytes per second (or packet) sent by HostX to HostY

Table 2. DFMxO features [1].

Number	Features
1	no. of created (or used) AAA connections by HostX
2	no. of BBB connections created by HostX using any port (or PortV) to connect to any port (or PortW) on any other hosts
3	no. of TCP packets sent by HostX with FlagZ
4	no. of packets (or bytes) sent by HostX
5	Average no. of bytes per second sent by HostX
6	Average no. of bytes per packet sent by HostX
7	no. of ICMP destination unreachable packets received by HostX
8	no. of ICMP echo request (or reply) packets sent by HostX

3 Implementation

This chapter explains how our system allows on-line feature extraction from huge amount of packets. We choose C as developing platform and our underlying operating system is Ubuntu 12.04 because feature extraction module is expected to be added to Snort in the future. Snort is linux-based NIDS and is implemented in C. Moreover, *CSQL Cache*, quickly accessible memory for our system, also works under linux operating systems.

3.1 Architecture

Figure 2 represents an overall view of the top level view of our system. Our system allows selection of packet features in real time. The aim is to separate *Features Extraction Module* from *Data Reader Module* and *Statistics Extractor Module*, so it can be used as an independent module in NIDS [1].

Data Reader Module collects data from *tcpdump* files and identifies intrusions. Labeled intrusions are required in the module of extraction statistics [1].

The aim of *Time Reconstruction Module* is inter-arrival synchronization for packets. Consequently, it recreates timestamps. In this way, the implementation of the next module is the same for on-line data capturing and reading data from a tcpdump file [1].

Features Extraction Module extracts also BF and DF features. BF group of features are used to compute DF group of features and DF group of features have a potential for successful attack detection. There are four main groups of DF features: DFS, DFMxB, DFMxO and DFMxG features. We show two subsets of these features. Subset of DFMxB features is in Table 1 and subset of DFMxO features is in Table 2.

MySQL database is used in [1]. In our opinion, this kind of database is not entirely suitable for on-line detection. We use *CSQL main memory database*, which acts as cache between our system

and MySQL database. It can be used in two modes: unidirectional and bidirectional. We used only unidirectional mode from CSQL database to MySQL database. Inserts and updates to CSQL database are automatically appeared in MySQL database [3].

Statistics Extractor Module is used to mine derived features from MySQL database with regard to specific types of attacks. It acts in off-line mode. Relevant features are gained through this module and these relevant features can be used for successful attack detection.

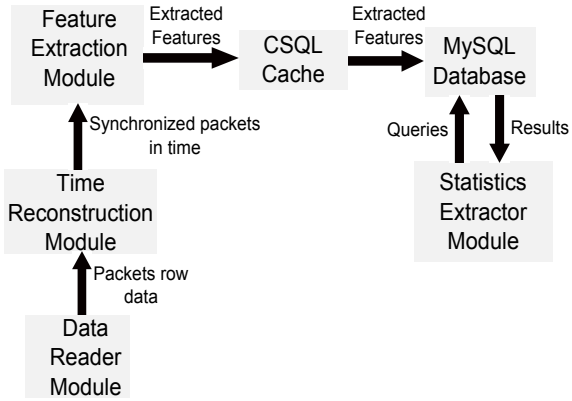


Figure 2. Architecture.

3.2 Feature Extraction Module

All features are grouped into five main groups as follows: *Packet data group*, *Host data group*, *Connection data group*, *HostX-HostY data group*, and *Network data group*. These five groups [1] allow the extraction of all the previously presented features and categories from the classification schema in Figure 1.

The Packet data group is used to store the BF category. A container is created for each encountered packet [1].

The Host data group is used to store any feature that is related to a particular host (i.e., DFMTO, DFMCO). A container is created for each individual host. Each host is uniquely identified by its source IP address [1].

The Connection data group is used to store any feature that describe a single connection (i.e., DFSTB, DFSTU, DFSLB, DFSLU). A container is created for each new encountered connection. Each connection is identified by a unique ID composed of 5 attributes as follows: (IP 1; IP 2; Port1; Port2; Protocol), where host IP1 uses Port1, host IP2 uses Port2, the connection is using Protocol, and IP 1 is lower than IP 2. In such a way, regardless of the packet direction, the connection ID will remain the same [1].

The HostX-HostY data group is used to store those features that characterize the exchanged information between HostX and HostY (i.e., DFMTB, DFMTB). A container is created on demand for each pair of hosts that exchange messages. This data group is uniquely identified by IP 1 and IP 2 (IP addresses for the two hosts), where IP 1 is lower than IP 2 [1].

Finally, *The Network data group* is used to store those features that are related to the network itself (i.e., DFMTG, DFMCG). The network is uniquely identified by the sniffer position [1].

4 Preliminary Results

The preliminary results in Table 3 are taken from [1]. They show that different groups of features are predisposed to highlight particular types of attacks.

Table 3. Preliminary results [1].

features	type of attacks or gained information
DFSTB and DFSTU	bursty attacks
DFSLB and DFSLU	stealthy attacks
DFSTB and DFSLB	attacks with one connection
DFMTB and DFMCB	vertical scanning attacks, DoS attacks
DFMTO and DFMCO	DDoS attacks, horizontal scanning attacks

5 Future Work

The aim of our system is to create the extension based on artificial intelligence of Snort. Snort is rule-based, pure software NIDS. Rules used in Snort enable quick detection of known attacks. Snort is easily extensible with new plugins [2].

However, new attacks would be undetectable, because rules for new attacks cannot be available. Using methods of artificial intelligence would improve Snort detection. Methods of artificial intelligence have to be used in real time. If not, it cannot be used effectively. Using methods of artificial intelligence in real-time is the reason for implementation of our feature extraction system.

6 Conclusion

Our system enables on-line features extraction and mapping to different types of attacks. Feature extraction is implemented in real time because of using cache between system and MySQL database.

Statistic results from database can be used to study features behavior versus different types of attacks. Preliminary results show that this system has the potential for mapping the network features into the network attack domain [1].

However, statistic results and application of relevant features in methods of artificial intelligence is currently not implemented and tested. A lot of work needs to be done here.

On-line packet processing is one of the most benefits of our system, which is the fundamental assumption for application of NIDS, e.g. Snort.

References

[1] Onut, I.V., Ghorbani, A.A.: Toward A Feature Classification Scheme For Network Intrusion Detection. In: *Proceedings of the 4th Annual Communication Networks and Services Research Conference*, IEEE, 2006.

[2] Sourcefire: Snort Users Manual. 2.9.3. edn., 2012.

[3] Sourceforge: CSQL Main Memory Database Cache. 2.1 edn.

Traffic Engineering Based on Statistical Modeling

Martin HRUBÝ*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
hruby@fiit.stuba.sk

Abstract. Quality of service is one of the main objectives when it comes to deploying sensitive applications into backbone networks. Applications like VoIP or streaming video are sensitive to network performance parameters which are subject to frequent change. In this paper we propose and implement a method for statistical modeling of network performance parameters. Based on this modeling we extend well known algorithms for finding shortest paths and create dynamic, reconfigurable traffic engineered paths. Our approach was implemented and verified in a laboratory environment. We provide measurement results and analysis.

1 Introduction

Rapid development of network applications in the last decade and their incorporation into daily lives of a significant portion of the population, have imposed a burden on existing underlying computer networks. The Internet was designed with best effort service in mind where connectivity was the most important issue. Today, connectivity is taken for granted. New approaches and technologies had to be implemented to accommodate the various applications and their diverse requirements. Bandwidth over-provisioning though still widely implemented, does not solve the issues of time-sensitive network traffic with voice or multimedia content where traffic prioritization must be deployed. To address these issues, various QoS mechanisms where proposed and implemented. Over the last decade it became apparent that traffic engineering would play a major role in the effective and efficient management of the installed capacity in large-scale networks [1]. We observe a push toward better utilization of existing network resources on the part of service providers, who carry ever increasing loads of traffic on their backbone networks. This initiative termed "traffic engineering'' is proving to be advantageous from both a QoS perspective and an economic perspective. Traffic engineering encompasses approaches to optimal resource utilization in computer networks while preserving quality of services. In recent past, there have been traffic engineering extensions to both routing protocols to incorporate traffic load in the link state advertisements and path selection decisions [2, 3, 4]. These solutions are applicable to IP networks, but do not offer the scalability achievable in MPLS networks. With the rapid

* Doctoral degree study programme in field: Applied Informatics
Supervisor: Assoc. Professor Margaréta Kotočová, Institute of Computer Systems and Networks, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 343–350.

incorporation of MPLS into large-scale service provider networks, traffic engineering capabilities were soon delivered as well. This however meant that existing underlying technologies had to be extended in order to provide the necessary support for signaling, path establishment and routing, see [5, 6, 7, 8]. These extensions enabled the signaling, path setup and maintenance of traffic engineered paths (*either explicitly defined, or determined by a constraint-based routing protocol*) in MPLS. Most approaches focus on finding paths with available (*reservable*) bandwidth in order to spread the traffic load more evenly across all available network resources. Modern multimedia applications like VoIP however require discreet, predictable values of time parameters, like delay and jitter as opposed to bandwidth. Routing delay and jitter sensitive traffic across traffic engineered paths setup to optimize the load of bandwidth may lead to suboptimal QoS/QoE [9].

Our objective in this paper is to present an implementation of our proposed n-CUBE model [10] for statistical modeling of network performance parameters. We have made a theoretical contribution to the area of active network measurements, but thus far our research was focused on the design and we have only gathered simulation results. Recently our model was implemented into a real application suite (*see further chapters*) and tested in a real network environment. This implementation is able to differentiate traffic flows and optimize the flow of traffic as result of changes in MPLS networks. Details about the statistical modeling and path determination can be found in [9, 10].

2 Concept

In this section we provide a brief overview of our proposal. This includes the logical elements and processes which are part of our optimization system. The n-CUBE modeling [10] is only a part of the entire suite. Other features include visualization (a function of the Matlab suite, which was modified and adjusted for our means), data storage, retrieval of network statistics and configuration update. All of these elements are depicted in Figure 1 below.

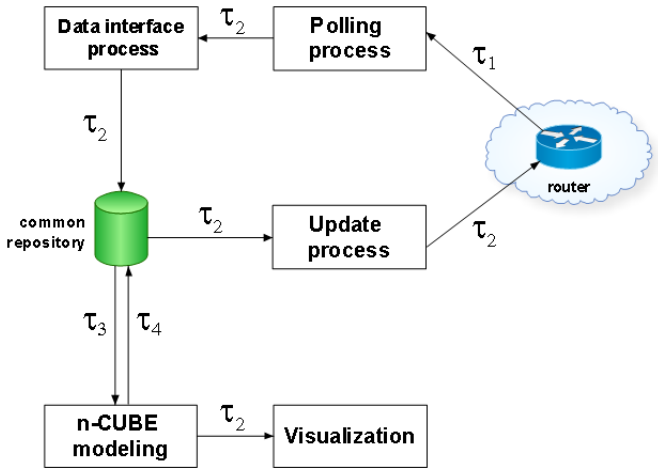


Figure 1. Logical functional elements of our model with parameters.

To briefly summarize, the network is observed and defined network performance parameters are gathered at specific intervals, and then stored in a common repository. There, they are available to the n-CUBE model which, at specified intervals, retrieves data stored in the common repository and performs statistical modelling and best-path search. The best paths are able to be visualized. A configuration modification is then fed back to the repository where it is prepared for delivery back to the network, thus creating the traffic engineered path.

Table 1. Description of the parameters of our model.

parameter	description
τ_1	*Defines a period of time after which data retrieval from the network takes place, in seconds*
τ_2	*Defines a period of time after which and action is triggered, in seconds (zero is instantaneous)*
τ_3	*Defines a period of time after which new data is fed into the n-CUBE for modelling , in seconds*
τ_4	*Defines a period of time after which new metrics are fed back into the network, in seconds*

We provide an overview of the logical functional elements of our model together with parameters. Our model can be divided into several functional processes, each responsible for a different function.

- Polling process – responsible for data retrieval from routers. This process periodically accesses network nodes every τ_1 seconds and retrieves measured IP SLA performance parameters for each link
- Data interface process – responsible for storing the measured data in a common repository where it is readily available. This process is triggered automatically once the Polling process retrieves all the information and the data interface process immediately stores the data to the common repository (given that the τ_2 time is equal to zero)
- Update process – responsible for retrieving configuration updates from the common repository as soon as they're available (given that the τ_2 time is equal to zero) and immediately pushes the changes to affected routers
- n-CUBE modelling – responsible for retrieving network performance parameters from the common repository every τ_3 seconds, feeding this information into the CUBE and assigning links to areas, thereby also assigning metrics. The result of modelling is fed back into the common repository every τ_4 seconds (this includes mostly link metric updates)
- Visualization – this is a by-product of n-CUBE modelling, where a graph representing (a portion of) the managed network together with calculated shortest paths can be seen. The visualization is updated every τ_2 seconds (instantaneous if τ_2 is set to zero).

3 Implementation

In this section we briefly describe the implementation of our proposed n-CUBE model. For implementation purposes we have decided to break up functional elements into blocks. The blocks will function as separate entities with possible multiple instances per block. Blocks are depicted in Figure 2 and defined as follows:

- Application in the NMC – this application leverages a common existing infrastructure of the network management centre which has access to all managed routers from a set of centralized jump servers (*i.e. Pollers*). The application will feature three autonomous processes working asynchronously and achieving:
  - Polling – fetching runtime variables from routers via SNMPv3 (IP SLA statistics, list of interface, routing table, LFIB)
  - Update – pushing updated configurations to routers via SSH (modification of link weights as a result of n-CUBE model weight assignment)
  - Data interface – storing the fetched runtime variables into a common repository, which is accessible to all (e.g.: file descriptor, SQL database, etc.)

- Common repository – this should be a universally, but securely accessible resource which enables fast data storage and retrieval. Consistency of data and shared access is guaranteed by underlying methods (operating system or database system).
- n-CUBE modelling – this application will retrieve runtime data stored in the common repository and model link weights based on multivariate normal distribution of desired network performance parameters. This will be achieved by a Matlab application.

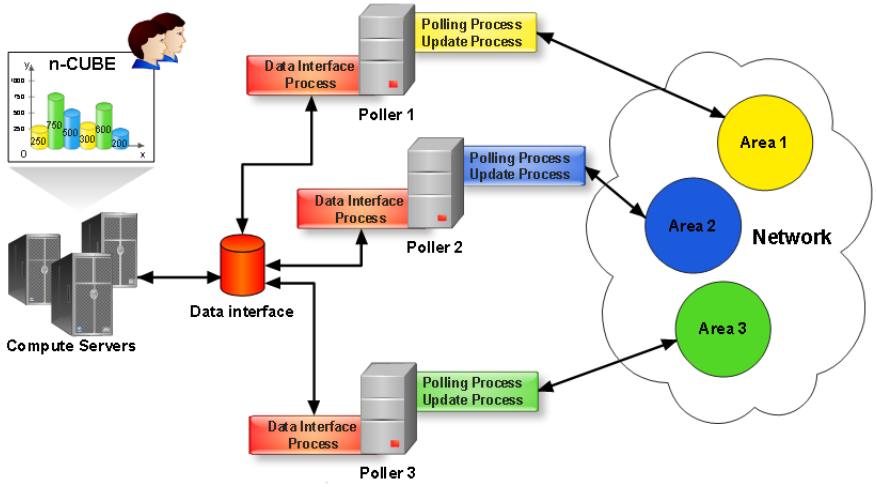


Figure 2. Data flow and segmentation of the network and system.

For verification purposes, the entire solution will be deployed in a network laboratory environment. Our design supports clustering and segmentation and thus enables scalability for long-term use. The laboratory environment designed for our experiments is depicted in Figure 3.

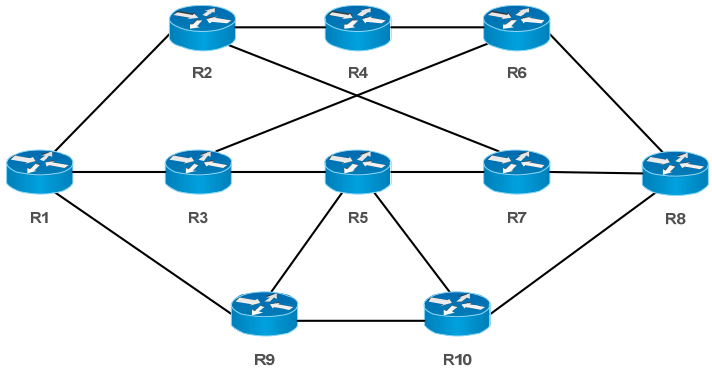


Figure 3. Physical topology for use in our experiments.

Traffic will be artificially generated into our network topology by use of the Iperf network traffic generator to and from end workstations. Network performance parameters will be evaluated on each link (*by IP SLA probes*) and in addition also end-to-end (*by IP SLA probes from routers SLA-SENDER and SLA-RECEIVER*) to determine improvement, see Figure 4.

Inside of the cloud is an MPLS network and all the routers are under management of our optimization system. We will conduct two experiments (*detailed in the following chapter*) where

we will artificially generate network traffic and mark it (*in IPv4 ToS field and MPLS EXP field*) as required by the experiment.

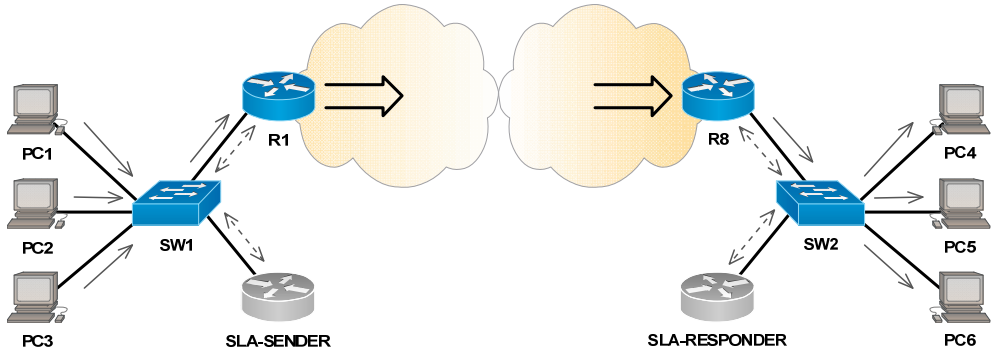


Figure 4. Traffic influx and outlet.

4 Results

In the first experiment, IP SLA measurement probes will be sent (in addition to being sent on every link as part of data gathering) end-to-end to quantify end-to-end performance parameters between communicating parties. All generated traffic except the IP SLA probes will be part of the same class and bears the same marking (in the MPLS domain it will be EXP=1). The IP SLA probes will be sent with a different marking (in the MPLS domain it will be EXP=2). Traffic optimization will therefore be done for all IP SLA traffic (EXP=2). One-time optimization option was used.

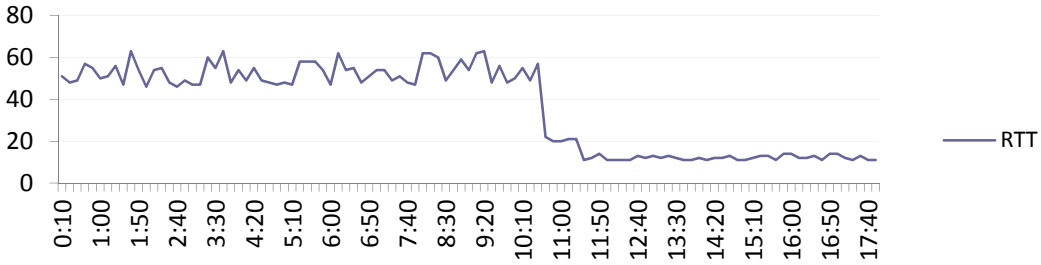


Figure 5. Experiment 1, measured round-trip time in milliseconds (end-to-end).

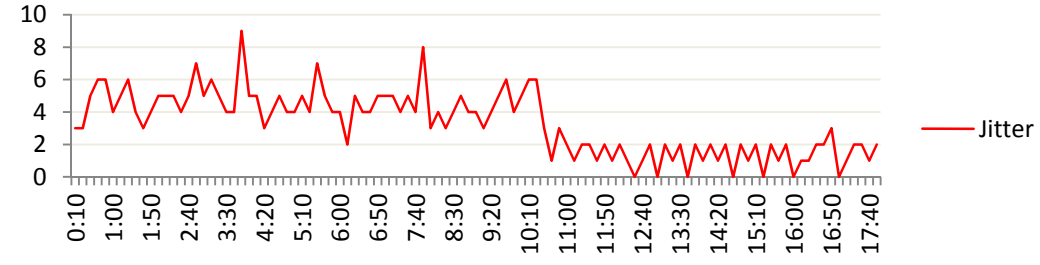


Figure 6. Experiment 1, measured jitter in milliseconds (end-to-end).

Table 2. Timeline of Experiment 1.

parameter	description
0:00	*PC1, PC2, PC3 start generating traffic at maximum possible rate*
10:00	*Traffic optimization is triggered*

We have generated traffic and IP SLA probes (end-to-end) with different markings, but prior to optimization only the routing protocol was used to determine the shortest path to the destination. This means that all traffic was forwarding along the same path, regardless of the marking. Once the optimization option was triggered, the IP SLA traffic (with a different marking, MPLS EXP=2) was suddenly being routed over a different path. Thus the measured end-to-end delay and jitter fell dramatically. In this way we optimized EXP=2 traffic and offloaded it from a congested path, thus improving the overall perceived quality of service in that traffic class. In the following Figure 7, we provide visualization, which is part of our optimization implementation. Prior to optimization, the default path (as chosen by the routing protocol) was 1 -> 3 -> 6 -> 8. And this was the path that all traffic was forwarded along. Once the optimization was triggered, a new separate path was found based on the metric assigned dynamically in our n-CUBE model. A traffic engineering tunnel was setup along the determined path and traffic marked as EXP=2 in the MPLS domain was forwarded along this path. The new found (optimized) path can be seen in Figure 7 on the right.

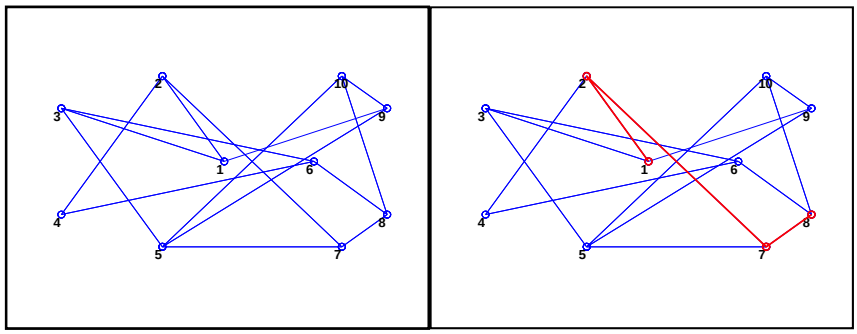


Figure 7. Visualization of the traffic engineered path (on the right).

In the second experiment, we will send two distinct IP SLA measurement probes. One probe will be marked as EXP=4 in the MPLS domain and the other as EXP=5. Traffic generated from PC-1 to PC-4 will be marked as EXP=5 in the MPLS domain and all other generated traffic (between PC-2 and PC-5, and PC-3 and PC-6) will be marked as EXP=4. Traffic optimization will be done for traffic marked with EXP=5 and a class-based tunnel will be created to transmit all traffic marked as such. All other traffic will be sent over the original path (before or after optimization).

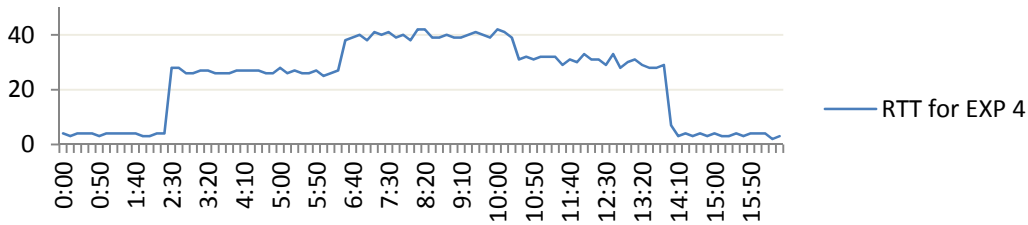


Figure 8. Experiment 2, measured round-trip time in milliseconds (end-to-end, EXP=4).

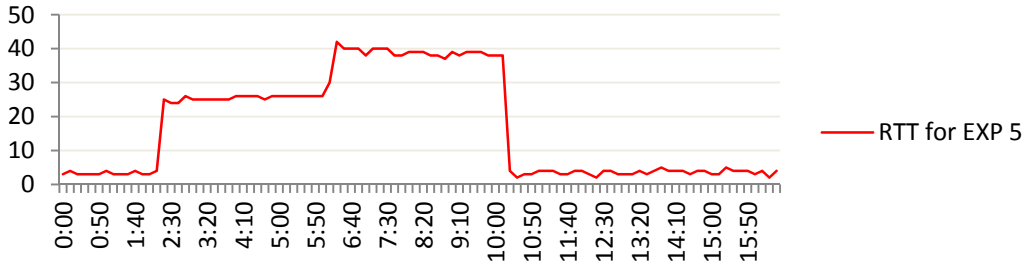


Figure 9. Experiment 2, measured round-trip time in milliseconds (end-to-end, EXP=5).

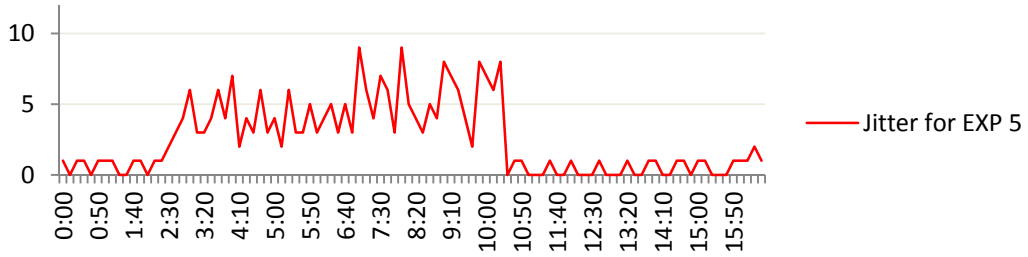


Figure 10. Experiment 2, measured jitter in milliseconds (end-to-end, EXP=5).

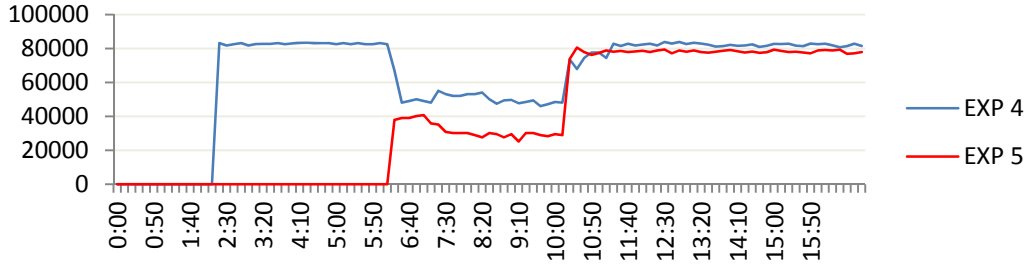


Figure 11. Bandwidth of generated traffic in kbps (both classes of traffic).

Once the optimization was triggered, traffic marked as EXP=5 in the MPLS domain was forwarded along a new, traffic-engineered path. Thus we have redirected the flow of traffic away from the congested path. As can be seen in Figure 11, once EXP=5 traffic was redirected to a new path toward its destination, the throughput improved also for the rest of traffic (marked EXP=4).

Table 3. Timeline of Experiment 2.

parameter	description
2:00	*PC2, PC3 start generating traffic marked as EXP=4 at maximum possible rate*
6:00	*PC1 starts generating traffic marked as EXP=5 at maximum possible rate*
10:20	*Traffic optimization is triggered*
17:00	*PC1, PC2, PC3 stop generating traffic*

5 Conclusion

In this paper we have extended our ongoing research and implemented our proposed n-CUBE model in a real network laboratory environment. We conducted two laboratory experiments. The experiments have shown that our model is capable of finding traffic engineered paths in MPLS networks suitable for traffic which is sensitive to the observed parameters – in our case delay and jitter. One-time optimization was used and it achieved dynamic traffic rerouting away from a congested path in reasonable time. Further experiments will be carried out with our model and we will publish results of continuous and progressive optimization options.

Acknowledgement: The support by Slovak Science Grant Agency (VEGA 1/0676/12 "Network architectures for multimedia services delivery with QoS guarantee") is gratefully acknowledged.

References

[1] Awduche, D.O.; Jabbari, B.: Internet traffic engineering using multi-protocol label switching (MPLS). In *Computer Networks: The International Journal of Computer and Telecommunications Networking*, vol. 40, no. 1; 2002; ISSN: 1389-1286; pp. 111–129.

[2] Katz, D.; Kompella, K.; Yeung, D.: *Traffic Engineering (TE) Extensions to OSPF Version 2*; RFC 3630; September 2003.

[3] Li, T.; Smit, H.: *IS-IS Extensions for Traffic Engineering.* RFC 5305; October 2008.

[4] Xu, K.; Liu, H.; Liu, J.; Shen, M.: One More Weight is Enough: Toward the Optimal Traffic Engineering with OSPF. In *Proceedings of the 2011 31st International Conference on Distributed Computing Systems (ICDCS)*; 20-24 June 2011; pp. 836–846.

[5] Awduche, D.O. et al.: *RSVP-TE: Extensions to RSVP for LSP Tunnels*; RFC 3209; December 2001.

[6] Jamoussi, B. et al.: *Constraint-Based LSP Setup using LDP*; RFC 3212; January 2002.

[7] Dana, A.; Khademzadeh, A.; Hoseinzadeh, H.: Selection of backup LSPs in MPLS network based on QoS. In *Proceedings of the 2009 11th International Conference on Advanced Communication Technology (ICACT 2009)*; vol.01; 15-18 February 2009; pp. 611–614.

[8] Figueiredo, G.B.; da Fonseca, N.L.S.; Monteiro, J.A.S.: A minimum interference routing algorithm. In *Proc. of the IEEE Int. Conf. on Communications*, vol.4, 2004, pp. 1942–1947

[9] Hrubý, M., Olšovský, M., Kotočová, M.: Solving VoIP QoS and Scalability Issues in Backbone Networks. In *IAENG Transactions on Engineering Technologies, Lecture Notes in Electrical Engineering*, Vol. 229, 2013, ISBN 978-94-007-6189-6, pp. 814.

[10] Hrubý, M., Olšovský, M., Kotočová, M.: Routing VoIP traffic in Large Networks. In *WCE 2012 World Congress on Engineering*, 4-6 July, 2012 Imperial College London, London, U.K, Vol. II. - Hong Kong : International Association of Engineers, 2012. ISBN 978-988-19252-1-3. pp. 798–803.

Efficient Repair Rate Estimation of Redundancy Analysis Algorithms for Embedded Memories

Štefan KRIŠTOFÍK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
kristofik@fiit.stuba.sk

Abstract. One important feature of redundancy analysis (RA) algorithms is repair rate. To estimate repair rate of various RA algorithms, software simulations of the algorithms on a number of fault memory maps representing real faulty memories are needed. In order to obtain realistic estimations, the fault distribution in maps has to resemble distributions observed in real chips as much as possible. In this paper, we show how fault distributions affect repair rate of some RA algorithms. Also, we propose a universal fault map generator based on random and cluster-oriented approaches suitable for repair rate estimations for RA algorithms.

1 Introduction

According to Semico Research Corp. forecast [1], area occupied by embedded memories on systems-on-a-chip (SoC) designs is slowly growing and will approach 70 % in the next few years. SoCs are moving from logic dominant to memory dominant. Overall SoC yield is therefore dominated by memory yield. As we move deeper into nanometer technology, embedded memory density and capacity grows which results in higher susceptibility of memories to various defects causing memory cells to perform faulty. This in turn causes memory (and SoC) yield to decrease. Maintaining acceptable yield has become an important task.

Built-in self-repair (BISR) techniques based on using redundancy are widely used to improve yield. Redundant rows and columns are added to the memory. Faulty memory cells are replaced by redundant ones. One important part of BISR responsible for finding a repair solution for memories is redundancy analysis (RA) algorithm. Recently, many BISR approaches and RA algorithms for various memory and redundancy architectures were proposed [2-6], [10-16]. One important feature of RA algorithms is repair rate (RR) defined as follows [4]:

$$repair\ rate = \frac{\#\ of\ good\ memories\ after\ BISR}{\#\ of\ total\ memories} \quad (1)$$

Repair rate depends on the number of redundancies available and effectiveness of RA algorithm. To estimate the RR of RA algorithms, a typical approach is to develop a software simulation tool

* Doctoral degree study programme in field: Applied Informatics
Supervisor: Assoc. Professor Elena Gramatová, Institute of Computer Systems and Networks, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 351–358.

capable of generating fault memory maps (also termed memory maps or fault maps) and executing the RA algorithm. Memory maps model a real memory as a two dimensional array of cells arranged into rows and columns. Examples of fault maps can be found in section 3.

In general, faults can be distributed across the memory map in various ways. To obtain realistic estimates of repair rates of RA algorithms, simulations need to be performed on a certain (usually high) number of memory maps with fault distributions resembling distributions seen in real faulty memories as much as possible. Wafer maps with locations of faults were previously difficult to obtain, but new techniques were introduced as early as late 80's [7]. These techniques showed that faults typically are clustered, not randomly distributed on wafer level. Many other studies (e.g. [8, 9]) confirm this observation. As there are many memory chips per wafer, this clustered distribution affects memory chips in such a way that some chips are fault free but others, located around the clusters have more faults (see Figure 1). Figure 1 depicts two examples of wafer maps with defect locations. The first example (a) assumes a very dense defect distribution whereas in the second example (b) the fault clusters occur mainly around the edges.

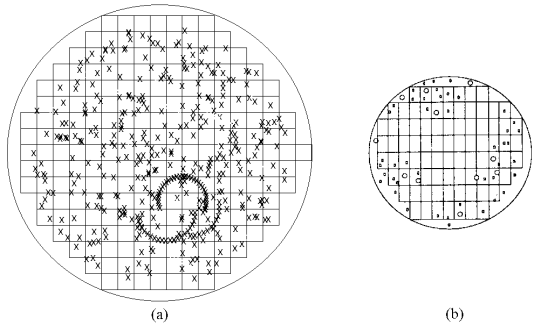


Figure 1. Wafer level defect distribution examples: (a) [8], (b) [9].

To simulate distributions such as in Figure 1, more sophisticated fault distributions than random have to be considered in simulation tools and yield models [8, 9]. On memory level, however, software tools able to simulate fault clustering that corresponds to wafer level defect distributions such as in Figure 1 are needed to estimate repair rates of RA algorithms in case the associated BISRs are used in real applications.

In this paper, we propose a universal fault memory map generator suitable for efficient estimation of repair rates of RA algorithms. It is based on random and cluster-oriented approaches.

2 Related work

The repair rates of RA algorithms are estimated in various ways. Usually, the authors implement their own software simulation tool capable of running the algorithm or in some cases more types of algorithms. Table 1 summarizes various approaches for RR estimations found in literature.

Faults injected into memory maps are usually of the various types. Single faults are most common. Usually 50 % or more of all faults in generated memory maps are single faults. Single fault is the only fault on its row and column and is a direct opposite of clustered fault. There are also many other types of faults injected into the fault maps. The fault distributions in fault maps used for RR estimations are either generated randomly or based on some theoretical distributions. The average numbers of faults in maps are varied. In some cases, they are set low, but there are cases where they are set as high as 100 per MB or even more. Fault maps are usually of various shapes and sizes (dimensions) up to 64 MB (8192x8192). The numbers of redundancies (R=rows C=columns in Table 1) are either set to a fixed value or experiments are conducted with varying numbers (up to 32 rows and columns per MB).

Table 1. Estimation of repair rate of RA algorithms.

	year	tool name	avg. faults	fault distribution(s)	# fault maps	fault map size(s)	# redund.	single faults %
[10]	2003	BRAVES	-	Poisson + Gamma	1552	1024x64	R 6-10 C 2-6	-
[2]	2006	-	17	random, adjustable % of 3 fault types	-	1024x64	-	adjustable
[11]	2006	-	83 to 189	random + Poisson	500	1024x1024	R 10-32 C 10-32	-
[12]	2006	eval. & verify platform	max. 10	Poisson	500	4096x128	-	0 % 50 %
[6]	2007	-	1-15	random	3000	1024x1024	R 2-5 C 2-5	20-65 %
[13]	2007	-	5-400	fixed % of each of 15 types of faults	18	32x32 – 8192x8192	R 1-30 C 1-30	-
[4]	2009	RepairSim	1-18	random	900000	1024x1024	R 5 C 5	69,32 %
[3]	2009	-	15	negative binomial	-	1024x1024	R 4-8 C 4-8	70 %
[14]	2011	-	7,8 3,3	Poisson Poisson	- 453	256x32 8192x64	R 3-6 C 3-9	20-100 % 70 %
[15]	2011	-	max. 10	random,	500	8192x64 32768x64	R 0-4 C 0-4	40-100 %
[16]	2011	-	-	fixed % of each of 4 types of faults	1000	1024x128 2048x64	R 1-4 C 1	0-80 %
[5]	2012	eval. & verify platform	max.10	Poisson	3	512x1024	R 1-5 C 1-3	-

3 Proposed fault map generator

The proposed fault map generator RNDCLUS is based on the random cluster generator approach proposed in [7], which is able to generate symmetric clusters of faults, using symmetric Gaussian distribution, on the wafer level. The clusters are centered in the center of the fault maps. In next step, it randomly stretches, rotates and relocates the clusters. In the last step, it adds additional clusters to the map that simulates scratches that occur during manufacturing process. We adopt this approach and use it on the memory fault map level. We however, omit the scratching simulation, but add an option to generate fault maps randomly when desired by the user. We now describe the fault map generation process of RNDCLUS.

3.1 Centered clusters and random option

Probability of fault occurring in memory cells is defined as follows [7]:

$$P(x,y) = Ce^{-(x^2+y^2)/2\sigma^2} \tag{2}$$

where C is a constant and σ is the standard deviation. The values of P(x,y) range from 0 to 1. The address values of x and y both range from -1 (for leftmost column address and uppermost row address) to 1 (for rightmost column address and lowermost row address).

To generate actual fault maps, for each map location, an auxiliary value of N(x,y) ranging from 0 to 1 is randomly generated. Then if N(x,y) < P(x,y) a fault is injected into the location given by corresponding values of x and y. The result is a symmetric cluster of faults around the center of the fault map. The value of σ sets the radius of the cluster and C sets the fault density within the cluster. An example of a fault map with a symmetric cluster is shown in Figure 2 (b). The fault clustering can be seen around the center as well as other faults near edges. An example of a fault map created with random option is shown in Figure 2 (a). The addresses of faults are

generated randomly and range from 0 to dimension-1. Random option is used exclusively with centered cluster function i.e. fault map either has a centered cluster or it is generated randomly.

3.2 Relocated clusters

Relocating the clusters of faults is done by generating a random values x_m and y_m. Their values range from 0 to dimension-1. Then all faults are relocated to a new location given by summing their original location (row address y, column address x) with the values x_m and y_m:

$$x = x + x_m \tag{3}$$

$$y = y + y_m \tag{4}$$

In case the new location is out of the bounds of the fault map, the approach [7] used the cropping technique and discarded out-of-bounds faults. We however modify this behavior and treat the fault map as a surface of a sphere and the fault re-emerges on the other side of the fault map. This is done to avoid possible high fault count losses in memory maps.

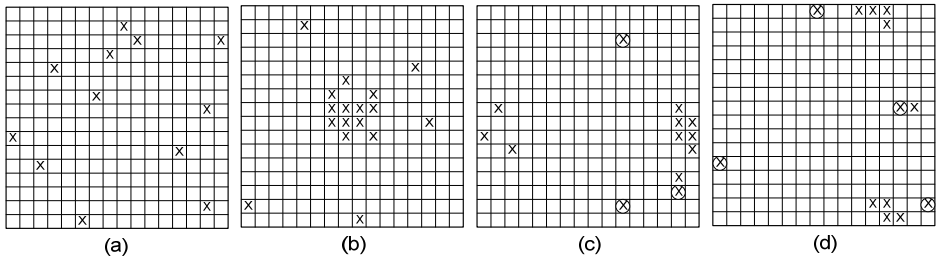


Figure 2. Examples of fault maps: (a) random, (b) centered cluster, (c), (d) randomized clusters.

3.3 Shaped clusters

Shaping of clusters is done by generating a random values x_s and y_s. Their values range from 0,1 to 1 meaning that cluster is stretched by a minimum of 0 % (when x_s or y_s=1) and up to 90 % (when x_s or y_s=0,1). Next, all faults have their original location multiplied by the values of x_s and y_s:

$$x = x * x_s \tag{5}$$

$$y = y * y_s \tag{6}$$

By executing the previous procedure, the clusters would be not only stretched, but also slightly moved towards the upper left corner of the map since their actual row and column locations are decreased. Therefore, after the procedure, we compensate this by following modifications obtained with trial and error experiments:

$$x = x + dimension * (1 - x_s)^{\frac{1}{x_s}} \tag{7}$$

$$y = y + dimension * (1 - y_s)^{\frac{1}{y_s}} \tag{8}$$

3.4 Rotated clusters

Rotation of clusters is done by generating a random value of angle α ranging from 0 to 359. The clusters are rotated by this angle counterclockwise around the center of the map. If a fault is out of the bounds of the fault map, we again do not use the cropping technique, as stated in section 3.2, and the fault re-emerges on the other side of the map. Since we use the non-standard left handed Cartesian coordinate system to assign location (addresses) to faults, it is first necessary to temporary convert them to standard right handed system. Next, the center of the coordinate system is "moved" to the center of the fault map by temporary modifying the fault addresses. Without this

step, the rotation would be done around the lower left corner of the fault map (0,0) and not around the center. Now, we are ready for the actual rotation and all faults have their locations in the map recalculated according to these standard rotation equations:

$$x = x * \cos\alpha - y * \sin\alpha \quad (9)$$

$$y = x * \sin\alpha - y * \cos\alpha \quad (10)$$

In the last two steps, we revert back the two temporal changes made previously and reverting back to left handed coordinate system.

3.5 Randomized clusters with added random faults

By combining the procedures from sections 3.1 – 3.4, we randomize the resulting fault distribution even more. Lastly, to add some more final randomization to resulting fault distributions, we add a small number of faults at random locations. This number is generated randomly and its value range from 0 to 5 meaning that a maximum of 5 randomly located faults are added to the distributions obtained by procedures from sections 3.1 - 3.4. Two examples of randomized clusters with added random faults are shown in Figure 2 (c) and (d). For example, the fault map in Figure 2 (c) was obtained from the fault map in Fig. 2 (b) by using values α=124, x_s=0,28, y_s=0,52, x_m=8, y_m=1 and number of randomly added faults was 3. The circled faults are the ones added randomly.

The results from Figure 2 (c) and (d) are very similar when compared to results in [17] and [18]. Both studies show the random fault map examples similar to that in Figure 2 (a) and clustered fault map examples similar to those in Figure 2 (c) and (d).

3.6 Parameters

Based on the observations in section 2, we have set the basic parameters of RNDCLUS according to Table 2. It is able to generate a large number of square fault maps of sizes up to 1024x1024. We have chosen the Gauss distribution, because it generates sufficient "starting" clustering of faults and then we modify it (sections 3.1 – 3.4) and still are able to achieve similar results to those reported in [17] and [18]. Therefore there is no need to use more complex theoretical distributions.

Table 2. Basic parameters.

Fault map size	avg. faults	# fault maps	fault distribution
16x16	10	1-100000	Gauss + random
32x32 – 256x256	15		
512x512	16		
1024x1024	17	1-10000	

Table 3. Parameters C and σ.

dimension	8	16	32	64
C	1	1	0,4	0,05
σ	0,25	0,15	0,1	0,018
dimension	128	256	512	1024
C	0,05	0,05	0,05	0,05
σ	0,009	0,0045	0,0023	0,0012

Table 4. Advanced parameters.

parameter	range	description
cluster_chance	0-1	A prob. there is a cluster in fault map. If there is not, random option is invoked.
cluster_reloc	0-1	A probability that if there is a cluster in fault map, it will be randomly relocated.
cluster_shp	0-1	A probability that if there is a cluster in fault map, it will be randomly shaped.
cluster_rot	0-1	A probability that if there is a cluster in fault map, it will be randomly rotated.
rndcnt_max	0-5	Sets the maximum of randomly added faults in case there is a cluster in fault map.
rndcnt_max_nc	-	Sets the maximum of randomly added faults in case there is not a cluster in fault map. These values are fixed and are equal to 2*(avg. faults) column from Table 2.

The average number of faults for small memories (16x16) was set to 10. For all other fault map sizes it was set to 15. As can be seen in Table 2, for large maps (1024x1024), we were only able to approximate this number to 17. The approximations were done on a trial and error basis while setting the values of C and σ and running the simulations until desired average numbers were

obtained. The resulting parameters C and σ for each fault map dimension are listed in Table 3. Fault maps in Figure 2 were created using the parameters from Table 3 for dimension 16. The procedures from sections 3.1 – 3.4 are used randomly with a certain probabilities given by values in Table 4, for each generated memory map. Most of these values are user adjustable.

3.7 Function

The functional flow of RNDCLUS is shown in Figure 3. Output is stored into the specified text file containing generated fault maps in the form of a list of fault location addresses.

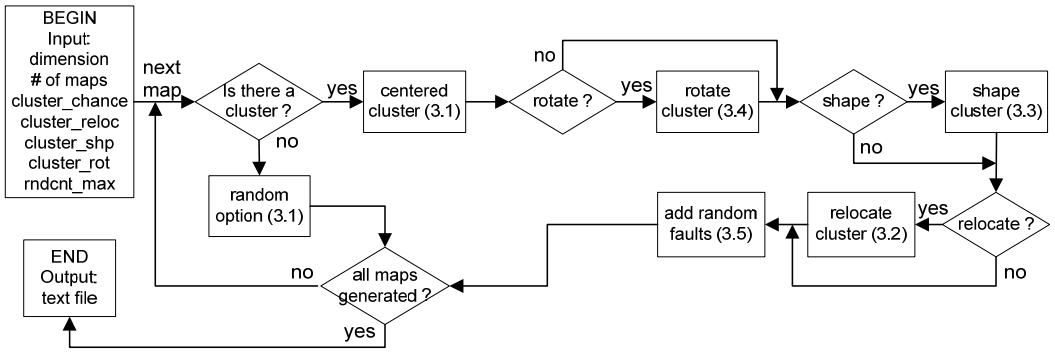


Figure 3. Flow diagram of RNDCLUS.

4 Experimental results

We now show how various fault distribution types can affect estimation of RR of RA algorithms. The MESP algorithm [3] was selected for implementation because it is targeted specially on cluster faults. We estimate the RR of MESP on small (dim. 16), medium (dim. 128) and large (dim. 1024) memories. Maximum number of generated maps from Table 2 was selected. The number of quadrants (sub-arrays of the map) of MESP is assumed to be 16. RNDCLUS generator is used in 6 various configurations shown in Table 5.

Table 5. Configurations of RNDCLUS.

configuration type	random	cluster-oriented				
configuration name	RND	C 0,75	C 0,5	C 0,33	add3	noadd
cluster_chance	0	0,75	0,50	0,33	0,75	0,75
cluster_reloc	-	0,75	0,75	0,75	0,75	0,75
cluster_shp	-	0,50	0,50	0,50	0,50	0,50
cluster_rot	-	0,75	0,75	0,75	0,75	0,75
rndcnt_max	-	5	5	5	3	0

We have selected these configurations to answer the following questions:

1. Is RR of MESP higher when dealing with clustered faults than with random faults, as is expected [3]? We observe the differences in RR between configuration RND and others.
2. How is RR of MESP affected by the percentage of clustered faults? We observe RR while decreasing parameter cluster_chance from 0,75 to 0,5 and then to 0,33.
3. How is RR of MESP affected by the number of randomly added faults? We observe RR while decreasing parameter rndcnt_max from 5 to 3 and then to 0.
4. Will RR of MESP estimated by RNDCLUS be similar to RR reported in [3]? If not, what are the possible causes?

Table 6 shows the repair rate of MESP using all 6 RNDCLUS configurations from Table 5. The number of redundancies ranged from 3 row and column blocks (3/3) to 12 rows and column blocks (12/12). In the last column, we compare the resulting RR obtained by RNDCLUS with the results reported in [3] where available. However, our results were obtained for average number of faults equal to 17 whereas results in [3] are for average number of faults equal to 15. Also, it is unknown how the numbers of faults were generated for each memory map (Were they generated equally or using some distribution?) and what was the total number of generated fault maps. By analyzing the results in Table 6, we are able to answer the aforementioned questions:

1. Yes. In small memories this becomes evident when the number of redundancies reaches 4 and for medium and large memories when it reaches 7.
2. RR slightly increases when the numbers of redundancies are small and it begins to decrease with increasing the number of redundancies. This is an expected result since the larger the map, the thinner are the generated clusters and the percentage of single faults increases which in turn has negative impact on repair rate.
3. RR increases greatly with all sizes of memories with decreasing the number of added random faults. This suggests that the initial value of rndcnt_max equal to 5 was set too high.
4. Yes, in most cases. Repair rates are similar to those reported in [3] when cluster-oriented distributions are considered. They are slightly lower with most of the RNDCLUS configurations. This may be caused by higher average fault count than in [3]. In case random option is used, the RR is significantly lower for any number of redundant blocks.

Table 6. Repair rate of MESP with different configurations of RNDCLUS.

dim.	# redund.	RND	C 0,75	C 0,5	C 0,33	add3	noadd	[3]
16	3/3	32,91	26,60	28,88	30,28	34,59	52,53	-
	4/4	45,22	52,90	50,43	49,02	64,28	77,33	-
	5/5	58,10	76,66	70,49	66,51	83,44	87,79	-
	6/6	71,77	89,71	83,74	79,98	91,70	92,43	-
	7/7	85,75	95,81	92,52	90,40	96,02	96,10	-
128	5/5	33,87	18,84	23,97	27,25	22,88	34,34	-
	6/6	40,88	34,03	36,31	37,88	41,39	55,44	-
	7/7	47,81	52,72	51,42	49,94	61,22	72,81	-
	8/8	54,81	69,69	64,75	61,26	76,29	83,26	-
	9/9	61,91	81,89	75,33	70,59	85,56	88,60	-
1024	6/6	36,35	27,68	30,73	31,36	34,77	49,32	-
	7/7	42,10	45,77	44,29	42,69	55,35	67,45	-
	8/8	47,96	63,10	58,14	54,13	71,60	78,85	65,50
	9/9	53,91	76,29	68,79	63,42	82,10	85,06	83,00
	10/10	59,62	84,45	76,09	70,36	87,54	88,25	93,00
	12/12	71,36	91,55	84,41	80,18	92,58	91,75	98,00

5 Conclusions and future work

The goal of this work is to offer the most exact estimations of repair rates of RA algorithms which can only be done if simulations are performed on memory fault maps that resemble fault distributions in real memory arrays as closely as possible. But to obtain such information from industry is not an easy task and we can only rely on other published approaches.

We reviewed the various known approaches to repair rate estimation problem and based on that, proposed a universal, user-adjustable fault map generator RNDCLUS. According to experimental results, it is suitable for estimation of repair rate of RA algorithms. By setting the values of various parameters of RNDCLUS, one can modify the output and is able to select whether the distributions are more random or more cluster-oriented.

Experiments have shown that the repair rate of RA algorithms is very heavily dependent on fault distributions in fault memory maps. Future research work will be invested to further study this dependency on other types of algorithms as well as to further improving the proposed generator with features such as adding new distributions or new cluster-generating approaches i.e. more than one cluster per map, cluster sizing, cluster positioning in quadrants and so on.

Acknowledgement: This work was partially supported by the Slovak Science Grant Agency (VEGA 1/1008/12).

References

[1] Semico: System(s)-on-a-Chip A Braver New World, Semico Research Corp. (2007). [Online; accessed February 18, 2013]. Available at: http://www.semico.com/press/press.asp?id=200

[2] S.-K. Lu, Y.-C. Tsai, C.-H. Hsu, K.-H. Wang, C.-W. Wu: Efficient Built-In Redundancy Analysis for Embedded Memories with 2-D Redundancy, *IEEE Tr. VLSI Systems,* (2006), vol. 14, no. 1, pp. 34–42.

[3] S.-K. Lu, C.-L. Yang, Y.-C. Hsiao, C.-W. Wu: Efficient BISR Techniques for Embedded Memories Considering Cluster Faults, *IEEE Transactions on VLSI Systems*, (2009), vol. 18, no. 2, pp. 184–193.

[4] W. Jeong, I. Kang, K. Jin, S. Kang: A Fast Built-in Redundancy Analysis for Memories With Optimal Repair Rate Using a Line-Based Search Tree, *IEEE Transactions on VLSI Systems*, (2009), vol. 17, no. 12, pp. 1665–1678.

[5] T.-J. Chen, J.-F. Li, T.-W. Tseng: Cost-Efficient Built-In Redundancy Analysis With Optimal Repair Rate for RAMs, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 31, no. 6, (2012), pp. 930–940.

[6] P. Öhler, S. Hellebrand, H.-J. Wunderlich: An Integrated Built-in Test and Repair Approach for Memories with 2D Redundancy, *Proc. of 12th IEEE European Test Symposium*, (2007), pp. 91–96.

[7] C. H. Stapper: Simulation of Spatial Fault Distributions for Integrated Circuit Yield Estimations, *IEEE Transactions on Computer-Aided Design*, (1989), vol. 8, no. 12, pp. 1314–1318.

[8] R. Allison: SAS/Graph Wafer Maps, in *Robert Allison's SAS/Graph Examples*, (2004). [Online; accessed February 20, 2013]. Available at: http://robslink.com/SAS/democd10/waferx.htm

[9] W. Kuo, T. Kim: An overview of manufacturing yield and reliability modeling for semiconductor products, *Proc. of IEEE*, (1999), vol. 87, no. 8, pp. 1329–1344.

[10] C.-T. Huang, C.-F. Wu, J.-F. Li, C.-W. Wu: Built-In Redundancy Analysis for Memory Yield Improvement, *IEEE Transactions on Reliability,* (2003), vol. 52, no. 4, pp. 386–399.

[11] H.-Y. Lin, F.-M. Yeh, S.-Y. Kuo: An Efficient Algorithm for Spare Allocation Problems, *IEEE Transactions on Reliability,* (2006), vol. 55, no. 2, pp. 369–378.

[12] T.-W. Tseng, J.-F. Li et al.: A Reconfigurable Built-In Self-Repair Scheme for Multiple Repairable RAMs in SOCs, *Proc. of IEEE International Test Conference*, (2006), pp. 1–9.

[13] S. Bahl: A Sharable Built-in Self-repair for Semiconductor Memories with 2-D Redundancy Scheme, *22nd IEEE Int. Symposium Defect & Fault Tolerance in VLSI Systems,* (2007), pp. 331–339.

[14] C.-L. Su, R.-F. Huang et al.: A Built-in Self-Diagnosis and Repair Design With Fail Pattern Identification for Memories, *IEEE Trans. on VLSI Systems*, (2011), vol. 19, no. 12, pp. 2184–2194.

[15] T.-W. Tseng, J.-F. Li: A Low-Cost Built-In Redundancy-Analysis Scheme for Word-Oriented RAMs With 2-D Redundancy*, IEEE Transactions on VLSI Systems*, (2011), vol. 19, no. 11, pp. 1983–1995.

[16] Y.-J. Chang, Y.-J. Huang, J.-F. Li: A Built-In Redundancy-Analysis Scheme for RAMs with 3D Redundancy, *Proc. of International Symposium on VLSI Design, Automation and Test,*(2011), pp. 1–4.

[17] A. Pelc, D. M. Blough: A clustered failure model for the memory array reconfiguration problem, *IEEE Transactions on Computers,* (1993), vol. 42, no. 5, pp. 518–528.

[18] M. Choi, N. Park, F. J. Meyer, F. Lombardi, V. Piuri: Reliability measurement of fault-tolerant onboard memory system under fault clustering, *Proc. of 19th Instrumentation and Measurement Technology Conference,* (2002), vol.2, pp. 1161–1166.

Power-Intent Integration into the Digital System Specification Model

Dominik MACKO*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
macko@fiit.stuba.sk

Abstract. Power consumption is becoming the key aspect in modern digital systems design. The current low-power design flow involves the application of some power-reduction techniques in an RTL (Register Transfer Level) or lower-level model. This paper describes a novel methodology for low-power design flow that supplements the existing one. It proposes an abstract form of the power-intent specification based on UPF (Unified Power Format) and integrates it into a system-level model. Power-intent specification at such an abstract level enables to manage power with higher efficiency, since power-aware decisions at this stage have more impact on eventual power consumption.

A paper based in part on this paper was published in Second Workshop on Manufacturable and Dependable Multicore Architectures at Nanoscale (MEDIAN'13).

* Doctoral degree study programme in field: Applied Informatics
Supervisor: Dr. Katarína Jelemenská, Institute of Computer Systems and Networks, Faculty of Informatics and Information Technologies STU in Bratislava

New Security Architecture for Mobile Data Networks

Martin NAGY*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
martinko.nagy@gmail.com

Abstract. Mobile wireless networks of GSM/GPRS/UMTS and lately LTE standards form the biggest global communication network worldwide. Despite its size, not much attention is dedicated to the security aspects of these networks. In this paper we propose a new security architecture for packet switched part of mobile network, which requires minimal changes in the existing infrastructure. New architecture enables network traffic filtering based on the end device demands. This solution is unique as the most of modern mobile platforms do not support firewall software at all and our approach enables this security feature to such devices.

A paper based in part on this paper was published in 11th Int. Conference on Advances in Mobile Computing & Multimedia (MoMM2013), ACM, pp. 253-259.

* Doctoral degree study programme in field: Applied Informatics
Supervisor: Assoc. Professor Ivan Kotuliak, Institute of Computer Systems and Networks, Faculty of Informatics and Information Technologies STU in Bratislava

Advanced Notification System for TCP Congestion Control

Michal OLŠOVSKÝ*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
olsovsky@fiit.stuba.sk

Abstract. Network throughput increase is usually associated with replacement of communication links and appropriate network devices. However it is necessary to bear in mind that effective and less intrusive increase of network throughput can be achieved via the improvement of existing protocol stack, mostly at network and transport layer. In this paper we propose an advanced notification system for TCP congestion control called ACNS. This new approach allows TCP flows prioritization based on the flow age and carried priority. The aim of this approach is to penalize old greedy TCP flows with a low priority in order to provide more bandwidth for young and prioritized TCP flows while providing more accurate details for loss type classification which is especially useful in wireless environment. By means of penalizing specific TCP flows significant improvement of network throughput can be achieved.

Amended version published in Information Sciences and Technologies Bulletin of the ACM Slovakia, Vol. 5, No. 2 (2013), pp. 39-43.

* Doctoral degree study programme in field: Applied Informatics
Supervisor: Assoc. Professor Margaréta Kotočová, Institute of Computer Systems and Networks, Faculty of Informatics and Information Technologies STU in Bratislava

Identification of Vulnerable Parts of Web Applications Based on Anomaly Detection

Rastislav SZABÓ*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
szabo@fiit.stuba.sk

Abstract. Apart from signature-based intrusion detection systems (IDS), anomaly-based IDS can be used for detection of attacks against web servers or web applications. Its detection mechanism is based on the identification of the HTTP requests that do not fit into the previously learned model of application's correct behavior. This paper describes a concept, which can be used for automated identification of vulnerable parts of web applications, based on the increased occurrence of anomalies in the production use of a web application. The output of our anomaly evaluation algorithm can direct security engineers and application developers to those modules of a web application, which are "attractive" for the attackers, or even point to some security vulnerabilities in particular modules of the application.

A paper based in part on this paper was published in 15th Int. Conference on Computer Systems and Technologies (CompSysTech'14), ACM New York, pp. 209-215.

* Doctoral degree study programme in field: Applied Informatics
Supervisor: Assoc. Professor Ladislav Hudec, Institute of Applied Informatics, Faculty of Informatics and Information Technologies STU in Bratislava

Improving Deployability of PKI in MANET Networks Routed by B.A.T.M.A.N. Advanced

Peter VILHAN*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
vilhan@fiit.stuba.sk

Abstract. This paper presents the partially evaluated concept designed to improve the PKI deployability in the mobile Ad-Hoc networks routed by B.A.T.M.A.N. Advanced. We have extended the B.A.T.M.A.N. Advanced 2013.1 routing protocol with authentication and authorization based on X.509 attributes certificates. Thanks to this modification we are able to mitigate various security risks and provide the more secure route for messages travelling through the network. As a result of our work we have tried to answer following questions: What is the performance impact of extended OGM message as the network grows and how well it scales? Does this concept provide us with the possibility of building the secure PKI infrastructure in the MANET environment?

1 Introduction

MANET networks are special kind of mobile Ad-Hoc network, which doesn't rely on fixed infrastructure. One of the main advantages is the ability to form the network in purely Ad-Hoc manner, without any costs spent on the infrastructure, like access points, antennas and so on. Nowadays we can found these kinds of networks in various conferences, or meetings, where group of people needs to exchange data or share the connection to the Internet.

In this paper we introduce our concept of Public Key Infrastructure, also known as PKI in this kind of networks. PKI consists of trusted third party – certificate or attribute authority – and clients which rely on it and trust to certificates signed by this authority. The common way of how certificate authorities work, is binding public key to legal identity. This way certificate authority confirms the identity of network entity. If we trust this certificate authority, we can safely communicate with any other node, which owns certificate issued by this authority. This concept is based on the security of private key used for the generation of certificates and works reliably in infrastructure networks, where certificate authorities need to fulfill strong security criteria. Despite of this, from time to time we can read about incidents leading to revocation of certificates. On the other hand, MANET networks are completely different story.

One of the way how MANET tries to mitigate attack is the network homogeneity. This means, that every node in the network should provide the same level of functionality like routing

* Doctoral degree study programme in field: Applied Informatics
Supervisor: Assoc. Professor Ladislav Hudec, Institute of Applied Informatics, Faculty of Informatics and Information Technologies STU in Bratislava

or provide the same set of services. This way an attacker should not be able to target its attack on a specific service or node.

The question is how to safely distribute certificate authority functionality to the nodes in the network, without compromising the security of its private key? Several approaches have been introduced during the last years, which try to cope with this problem in following ways:

- Partially distributed certificate authorities, introduced by Zhou and Haas [1] and Yi and Kravets [2], distributes the functionality of certificate authority to several nodes in the network. Each of these nodes generates just one part of the certificate. The main drawback of this concept was the presence of special node called merger, required for the construction of the certificate – this introduces the single point of failure
- In Fully distributed certificate authorities introduced by Luo et al. [3], each node shares the part of certificates authority private key and at least *t* nodes were required, to provide the functionality of certificate authority. The security of this concept depends on *t* value. Lower *t* value means better service reachability, but higher chance to compromise the certificate authority. On the contrary, if the value of *t* was too high and the node requesting services from certificate authority didn't have at least *t* neighbours, certificate could not be generated and the service was unreachable.
- The certificate chaining-based approach by Capkun et al. [4] was built on the chain of trust. Moreover various people have various level of security knowledge. Therefore the chain was as strong as its weakest point.
- There were several other approaches as Mobility based by Capkun et al. [5], benefitting from the fact, that node can move close to the certificate authority. Parallel by Yi and Kravets [6] combining known approaches, which could introduce new security threats as a result of combination. Cluster based by Ngai Xia in 2007, dividing the network into clusters and electing the only node in the cluster, responsible for the communication with other clusters. Xia, Wu and Chen introduced Identity based [7] authority, utilizing public key identity, optimised for the Optimised Link State Routing Protocol. Various other approaches like Grid based by Lee in 2007 or virtual authority based by Shukat and Holohan[8], were introduced, but none of them provides us with the successful solution usable in regular conditions.

Main problems of introduced solutions were caused by the absence of routing protocol, which could be used for safe communication during the process of the certificate authority establishment. Routing is critical part of the network and various attacks like Man-In-The-Middle, BlackHole attack or Sybil attack can be effectively performed in the MANET networks.

There are several modifications of well-known MANET routing protocols, like Optimized Link State Routing and Ad hoc On Demand Vector utilizing PKI, but at most cases the formation of certificate authority and distribution of keying material were required before the network can operate. On the other side, distribution of keying material couldn't be completed without routing protocol support. This was also known as chicken&egg problem.

2 Proposed solution

After the analysis we were able to identify critical parts of PKI security concept:

1. Preserve the maximum possible level of homogeneity in the network, to harden the targeting of the attack.
2. Design the way how to grant permissions to entities with high level of granularity. All of already introduced solutions provide the nodes with "all or nothing" level of PKI permissions used when accessing network resources.
3. Design the PKI architecture with failure in mind. One of the characteristics of the MANET is the fact, that we cannot guarantee overall and consistent security level over the network.

Each node is owned by different user with different security knowledge. Furthermore, the nodes are mobile, so can be lost, stolen or compromised in a shorter time. Important task is to shorten the time required for the detection of anomalies. This can be done with help of distributed intrusion detection system (DIDS).

4. To further mitigate security problems and chicken&egg phenomenon, the routing protocols should be PKI aware.

The proposed concept consists of the following parts:

- Due to significantly shorter time between the security incidents in MANET we have to use certificates with shorter validity. Furthermore, to be able to grant permissions to network resources with higher level of granularity, we have opted for use of attributes certificates. Attributes certificates are issued by attributed authority and have a lot shorter validity than general certificates issued by certificate authority. In some cases, we don't need to establish nor manage the certificate revocation list, further referred as CRL, because certificates will timed-out sooner than CRL could be fully distributed through the network. Last but not least, there are legal consequences, like impossibility to confirm the identity of node and its owner. Due to this, our concept binds node's identity to its public key, instead of its real identity.
- The main idea behind this concept is*: Let the node gets its digital identity and to build its reputation upon it.*
- One node can have several identities. Various attribute authorities can issue certificates for the same node. This way, node can limit the negative effect, if the issuer of its certificate was compromised and certificates had to be revoked.
- The reputation and permission of each identity is independent and not transferable between identities.
- As an solution to "all or nothing" problem, we have set up the following predefined levels of permissions, which node have to gain, before it becomes the fully integrated part of the network. We can think about this as about kind of "accession talks":
  - L1 – endpoint node, this is the basic permission level that node gets with the new certificate. With L1 permissions node cannot participate on a routing processes in the network, but can access some of the network services, defined by network security policy
  - L2 – if there is sufficient communication history between the node and the hosts providing services in the network and its identity isn't listed on the blacklist, node can ask the issuer to elevate permissions of its certificate. With this permissions level, node participates on the network routing, mediate certificates for the new, connecting nodes and run various services like distributed certificate storage and distributed intrusion detection system. Data storage both of these services are implemented via distributed hash table.
  - L3 – the highest level of permissions, node transform into attribute authority and cross certificates between both authorities will be created.
- Furthermore we have designed two levels of trust between the authorities:
  - L1 – authorities cross-sign the certificate of each other. This way, nodes with certificate issued by these authorities can verify the certificate of each other.
  - L2 – distributed certificate store and DIDS of both authorities are merged

This way each attribute authority can built its own ecosystem of issued certificates and trusted authorities in the network. The number of authorities inside the network is not limited. Each node made its own decision to provide or not to provide trusted authority services, which depends on the amount of free nodes resources. Every node can transfer itself into attribute authority and issue certificates. However, the usefulness of these certificates will depend on the amount of cross

certificates between the issuer and another attributes authorities. A lot of the cross certificates means higher chance to verify certificates issued by another authority.

Similarly the node can own as many certificates as it wish, but it must participate on various services like distributed certificate storage or DIDS, resulting from its privilege level. The more certificates with higher privilege level the node owns, the lower is an effect of authority breakdown.

Thanks to cross certification between the authorities, the max length of the certificate chain to verify is extremely short, namely:

Node Y needs to verify certificate of Node X:

1. Upon receiving the Node X certificate, Node Y make a lookup in distributed certificate storage of its mesh to find out, if there is a cross certification between the issuer of Node X certificate (AA1) and issuer of one of Node Y certificates (e.g. AA2).
2. If there is, Node Y will download the cross certificate, verify AA1 certificate and then verify the Node X certificate with the help of AA1 certificate

2.1 Structure of PKI in MANET

The following requirements must be fulfilled to fully establish PKI in MANET:

1. Services that can use X.509 attributes certificates for the authentication
2. Routing protocol that can take advantage of the presence of PKI

2.2 Protocol for communication between nodes

As we have stated before, the solid PKI aware routing platform is required for proper implementation of PKI in MANET. We have analysed various protocols and have decided to use B.A.T.M.A.N. Advanced. The following chapter describes the changes we have made to B.A.T.M.A.N. to take advantage of PKI support.

2.3 Modification to B.A.T.M.A.N. Advanced

The B.A.T.M.A.N. Advanced is Layer 2 routing protocol supported in Linux kernel since the version 2.6.38 onwards. Its primary goal is the simplicity of configuration. All we need to do is to activate it on selected interfaces and after few seconds we have routed network. As it is Layer 2 routing protocol, MAC addresses are used for the routing, so it is fully independent of the network layer protocol and we can use IP, IPv6, IPX, or any other protocol on top of it.

The network is created from the partial mesh of nodes. Each node knows about the existence of other node in the network and knows the direction (MAC address of the neighbour) to which forward the message. Unlike link state routing algorithm, the node is not aware about the overall network topology. This on the other hand greatly reduces the computational requirements, so B.A.T.M.A.N. Advanced can be used on embedded devices, too.

The network convergence is possible thanks to B.A.T.M.A.N. OGM messages, which distribute the information about the connected nodes across the network. The detailed operation of protocol is out of scope of this document, but we will introduce basic principles and several parts that had to be modified.

Each B.A.T.M.A.N. Advanced enabled node manages several data structures:

- Originator table – contains information about other nodes in the mesh, next-hop destinations to these hosts, including alternate next-hops and network quality value. In our concept mesh consists of nodes with at least L2 level of permissions.
- Local translation table (LTT) – contains the list of nodes which don't participate in the mesh and we are providing them with routing services. These nodes can communicate with the rest of the network only through ours node. Nodes with L1 permissions certificates are in LTT.

- Global transition table (GTT) – is created as result of LTT flooding through the network and provides the mesh nodes with the ability to find the other mesh nodes responsible for the routing of data to destination node.

It is important to note that routing decision is based purely on the information of concrete routing node and there isn't any way of how to force the path of message through the mesh. Every node makes its own routing decision, which cannot be directly affected.

One of the disadvantages of B.A.T.M.A.N. Advanced comes from its simplicity. B.A.T.M.A.N. is missing any form of routing updates authentication and overall security is let on upper layer protocols. This is where our solution comes in.

To be able to prevent following security problems:

- Man In The Middle attacks – through the exploiting of certification issue process
- Denial of Service attacks – over helming the certificate authority with tons of requests
- Various routing attacks like Black Hole attack or Sybil attack

Following changes were made to the B.A.T.M.A.N. Advanced concept.

2.3.1 Adding of authentication to B.A.T.M.A.N. routing updates

The main idea is to build a safer, more rigid mesh. B.A.T.M.A.N. Advanced enabled nodes exchange routing data through the OGM messages. We have extended B.A.T.M.A.N. the way that every OGM packet transmitting an update data has to be verified, before it will be processed. The verification process consists of validation of OGM message signature. Nodes with the certificates from the same issuer, or other issuer if theirs attribute authorities are cross certified, can verify OGM message signature and check if the peers permission are at least L2 level. As a precaution against various DoS attacks, OGM messages, which cannot be verified, are dropped. This is the way how nodes are conserving system resources and network bandwidth.

Every OGM message is SSL signed with one of the certificates that node owns. Certificates used for signing process are changed in a round-robin manner with each OGM message. Thanks to the omnidirectional Wi-Fi transmit profile this is not a problem and in finite time each of our neighbours receive the proper update. Receiving node identifies the proper peer's certificate thanks to attached hash of this certificate.

For the signing, 1024 bit RSA keys can be used, as the validity of certificate will be rather short, but we recommend the use of 256 bit Elliptic curve keys, if you want to use the same key for many certificates. The use of RSA keys in signing process, gives us overhead of 144 bytes, while 256 bit EC keys does not take more than 98 bytes. Standard OGM header has size of approx. 27 bytes and OGM message containing single entry has approx. 52 bytes. The overhead caused by PKI signature is rather big, but since B.A.T.M.A.N. Advanced version 2010.0, OGM aggregation is enabled by default, so we are able to send more than one entry in the OGM message. Preliminary testing shows that PKI overhead is acceptable.

The side effect of this solution was the creation of network fragments consisting of nodes which weren't able to verify each other's certificate. This was caused by the non-existence of cross certificates between the issuers, but it is perfectly correct situation, which we have to deal with. Moreover this problem was getting worse in highly mobile networks, like Vehicular Ad-Hoc or MANET consisting of highly mobile clients. The problem is that in this case fragments are fully separated and cannot communicate together. Moreover all fragments except the one that contains its attribute authority cannot further manage certificates and are dying.

2.3.2 Cluster glue

We have designed the concept allowing fragmented clusters to communicate again. The idea is to help overcome communication outages caused by the moving of nodes out of its clusters signal range. If the node has the certificate which can be validated by intermediate cluster between

fragmented clusters, it can use this intermediate cluster and tunnel the traffic to the remote cluster. The new name, "*Edge node*" is introduced, describing the node with at least L2 permission certificates, which is actively part of at least two meshes(clusters)-the node has to have at least two valid certificates from different authorities, each with at least L2 security level. Moreover only clusters containing at least two nodes or at least one non-mesh clients are announced.

The following figure introduces the simplified network topology:

- nodes B and D represents edge nodes,
- different colour represent ownership of a certificate issued by different authority,
- nodes A (dark grey, further referred as DG) and B (light grey, further referred as LG) are attributes authorities.
- As we can see DG is fragmented by sub-mesh consisting of nodes {B,C,D}.

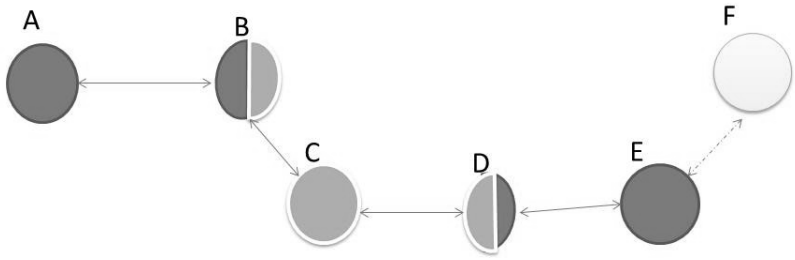


Figure 1. Example of fragmented network.

Cluster glue works the following way:

1. We have extended the format of an OGM message the way it can contains information about other meshes, too. The goal is to recognize edge nodes in the same clustered mesh(B, D). This way node D can promote itself as an edge node to DG and includes this information in OGM for mesh LG. Node B will later get this data and vice versa.
2. Node C receives broadcast containing OGM message from node D. The OGM was signed with certificate issued by LG, so node C can verify it. As a result, node C stores general information about node D in the Originator table and in the Global translation table. After the node C completes the OGM processing, it rebroadcasts it further to the network.
3. Node B receives the rebroadcasted OGM message, verifies it with LG certificate and continues in the same manner as node C. Furthermore, node B extracts information stating that node D is an edge node to DG sub-mesh and stores it in *"Edge node table".* We have extended B.A.T.M.A.N. Advanced with this table to store information about Edge nodes in the current sub-mesh. Edge node table contains data about remote cluster, edge node, link quality to edge node and the age of the entry, so old entries can be effectively identified.
4. The same process will happen from the opposite way (B → D)
5. After then, the network is ready to forward OGM for the fragments through the edge node. When the node B needs to send data to node E, it will look up next hop entry in originator table. Node B finds out that next hop to DG is D, reachable through the mesh LG. Node B will encapsulate the original DG packet into special OGM packet signed by LG and forward it towards the destination node D, node C.
6. When node C receives the OGM packet signed by LG, it verifies and processes it. The originators table contains node D, so node C can forward this message to the node D.
7. The edge node D receives the OGM packet, verify it with LG issued certificate, decapsulate DG signed data, verify it with DG and process it.

8. This way node E can mediate the new identity, signed by DG, for the incoming node F.

To limit the bandwidth used by the cluster glue, we have designed the following policy: In case an edge node (e.g. B) doesn't identify any remote cluster (e.g. DG), it reduces the rate of announcement itself as an edge node to the cluster (e.g. DG).

3 Performance measurements

As we have stated before, adding the PKI into OGM packet slightly limits the available space for the routing information. But thanks to the aggregation functionality, PKI will never use more than 12% of the available space, provided by MTU of size 1500 bytes.

The next question was, if there is any measurable impact of adding Edge node info into OGM message. We have made several simulations with various network topologies, consisting of 11-30 nodes and 3 to 10 attributes authorities.

On the Figure No.2, we can see dependency between the amount of certificates advertised on the edge node and the remaining space for the regular routing data in one OGM packet.

As we can see, there is plenty of free space in OGM packet and in general conditions (up to 5 certificates), majority of used space was allocated by PKI signature.

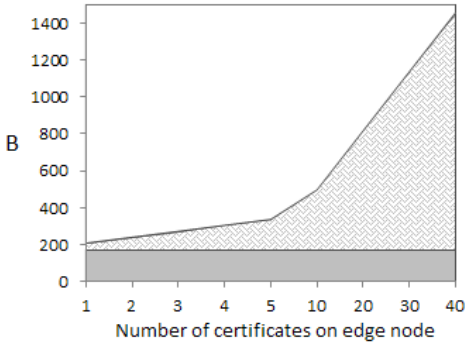


Figure 2. Allocated MTU space by Edge node crt.

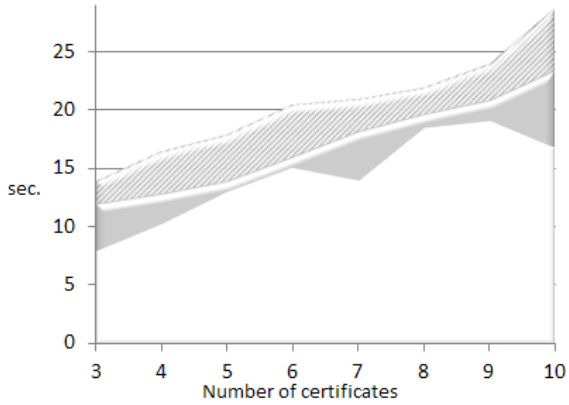


Figure 3. Network convergence time.

Furthermore we have set up our test environment consisting of the following components:

1. HP Micro Server N40L, 1,5 GHz, 2 cores, 2 GB RAM, running Ubuntu 12.04LTS, 32 bit
2. QEMU 1.2.2 + VDE switch 2.3.1 with colour patch, one instance per node
3. Wirefilter 2.3.1 for simulation of packet loss – one instance per connection between vde switches. Wirefilter is used to simulate the device movements and signal outages.
4. OpenWRT Barrier Breaker, trunk, mid-March revision and B.A.T.M.A.N. Advanced version 2013.1 from the OpenWRT repository.
5. PKI authority policy doesn't include data of DIDS, which was not implemented in time of analysis. The client certificate requests were satisfied according the pre scripted scenario.

As you can see on Figure 3 we have measured the network convergence time, in the network starting with 11 and finishing with 30 movable nodes. The question was, how significant will be the impact of edge nodes on the network convergence time. Delay could be caused by higher load associated with SSL operations, edge node detections and packet processing (encapsulation, decapsulation). The results vary according to actual network topology, but with the same topology we were able to get network convergence times within spread of no more than 25%. This means

that processing of edge nodes information doesn't have fatal impact on the convergence time and differences were caused by the random network topology.

4 Conclusion

We have presented the concept of public key infrastructure on top of B.A.T.M.A.N. Advanced routed network. The idea was to gradually raise nodes permissions to mitigate effect of MITM, Black Hole, Sibil Attack and compensate the fact, that each attribute authority is located on a single node. On the other side each node can create its own authority and build its own ecosystem of client nodes. Ability to verify certificate of other authority is based on the existence of cross certificates. This solution is more secure than certificate chaining and resource friendly, too. Shorter validity period of attributed certificated and the presence of DIDS, allows us to work without certificate revocation list. Furthermore we have designed the extension to OGM messages which brings authentication to B.A.T.M.A.N. Advanced. As a solution for fragmented network we have introduced the cluster glue, protocol extension which brings us back to game. As we have stated before the concept need to be further optimized, but proves that it work.

References

[1] Zhou, L., Haas, Z.Z.: Securing ad hoc networks. *IEEE Network*, (1999), vol. 13, no. 6, pp. 24–30.

[2] Yi, S., Kravets, R.: MOCA: Mobile certificate authority for wireless ad hoc networks. In: *Proceedings of the 2nd Annual PKI Research Workshop (PKI 2003)*. (2003).

[3] Luo, H., Zerfos, P., Kong, J., Lu, S., Zhang, L.: Self-securing ad hoc wireless networks. In: *Proceedings of the Seventh International Symposium on Computers and Communications (ISCC'02)*. (2002), pp. 567–574.

[4] Capkun, S., Hubaux, J., Buttyan, L.: Mobility helps peer-to-peer security. *IEEE Transactions on Mobile Computing*, (2006), vol. 5, no. 1, pp. 43–51.

[5] Cagalj, M., Capkun, S., Hubaux, J.: Key agreement in peer-to-peer wireless networks. *Proceedings of the IEEE* (Special Issue on Cryptography and Security), (2006), vol. 94, no. 2, pp. 467–478.

[6] Yi, S., Kravets, R.: Composite key management for ad hoc networks. In: *Proceedings of the First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous'04)*. (2004), pp. 52–61.

[7] Xia, P., Wu, M., Wang, K., Chen, X.: Identity-based Fully Distributed Certificate Authority in an OLSR MANET. In: *4th International Conference on Wireless Communications, Networking and Mobile Computing, WiCOM '08*. IEEE, (2008), pp. 1–4.

[8] Holohan, E., Schukat, M.: Authentication Using Virtual Certificate Authorities: A new Security Paradigm for Wireless Sensor Networks. In: *NCA '10 Proceedings of the 2010 Ninth IEEE International Symposium on Network Computing and Applications*. IEEE, (2010), pp. 92–99.

Extended Abstracts

An Approach to Crawled Data Semantic Annotation from Selected Domain

Filip BEDNÁRIK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
filip.bednarik@memes.sk

Extended abstract

This work presents an approach to semantic annotation of crawled data gathered by a crawler. It improves currently used approach [1] which for purpose of annotation uses two different kind of ontologies, the weak ontology in Java and the strong in OWL [2]. The new approach omits the first one, hence the semantic annotation method is more efficient to maintain, and moreover, it can benefits from various advantages offered by semantic web technologies such as data reasoning.

Webcrawler is a program which systematically visits web sources, where it gathers and stores desired data into database. In our case semantic database. We use Owlim [3] which heavily relies on ontologies describing entity relations and database structure. We need this ontology to create proper queries for database.

Currently used method uses two ontologies. One is used by crawler and one is used by database. The ontology in java has mappings from portal specific labels to classes and individuals in other ontology. So practically the same ontology is there twice in different technological spaces [4], one in OWL format and one in Java. This approach requires programmer to edit Java ontology when OWL ontology is changed and what is more, when portal changes or we want to crawl new portal, semantic web developer has to edit OWL ontology and programmer Java ontology.

In comparison to currently used method our method does not need conditions in parsing part. New approach reduces the number of lines in source file radically and easies whole implementation process. While crawling, program automatically detects type of property value and parses data accordingly. We can adapt to new portal very fast and we can easily distribute the tasks to semantic web engineer and programmer. Programmer is not bothered by data structure. Also code does not need to be changed when portal adds more values to one property. New approach also creates reports of missing properties in ontology. This is very convenient and with portal changes coming fast, programmer does not even have to change code in most of the cases.

Our method uses only single ontology which is defined in OWL files. These OWL files are standard ontology files. What is distinctive is that in order to use them as transformation model they must contain annotations. These annotations can be in different languages and suited for

* Bachelor degree study programme in field: Informatics
Supervisor: Dr. Miroslav Líška, Datalan, a.s., Bratislava

IIT.SRC 2013, Bratislava, April 25, 2013, pp. 399–400.

different portals. Each class, each dataproperty and each objectproperty must have its label so they can be used in transformation process.

Program reads these labels and creates map from labels to classes and properties which is then used in mapping process and query building.

However, this method usually cannot derive two properties out of one property. In that case we use semantic rules defined in .pie file. It is very convenient for semantic web developer because he has control over inference and source code does not contain semantic rules.

Advantages of the new approach:

- speed up crawling preparation process of new portals rapidly (approximately four times),
- increased consistency because of only one strong ontology,
- easier process of management,
- feature that creates list of labels and data types that can help us create ontology for specific portal,
- portals changes reports,
- the work between programmer and semantic web developer is properly divided,
- programmer usually does not need to change source code when portal changes,
- semantic web developer has more control over conditions and inferencing,
- increased effectiveness of work,
- decreased number of files needed in order to crawl portal,
- lower number of lines in source code needed to process document.

References

[1] Sestate Crawler, Datalan a.s. [Cited: February 19, 2013.]

[2] Web Ontology Language. [Online] [Cited: February 19, 2013.] http://www.w3.org/OWL/.

[3] Ontotext. Owlim documentation. [Online] [Cited: February 19, 2013.] http://owlim.ontotext.com/ display/OWLIMv53/OWLIM-SE

[4] I. Kurtev, J. Bezivin, and M. Aksit.: Technological Spaces: an Initial Appraisal. Industrial track, 2002.

Intelligent Control – "Camsoft"

Peter BRECSKA, Mário KUKA*

Secondary professional school
Františka Hečku 25, Levice, Slovakia
pbrecska@gmail.com

Extended abstract

Camsoft is a client-server system for the implementation of automatic measurements and also for quick and easy data mediation these measurements appropriately to user.

For the implementation of our project we had to tackle a variety of technical problems, directly or indirectly related to the proper operation of the system. We had to completely reconfigure an existing building safety camera system since it worked inefficiently and put a strain on the entire network (before our modifications, the flow of data over the network was over 3200 kbit/s, after our configurations adjustment and transition to new technology, the flow of data was only 160 kbit/s). For this we used AXIS Media Control software [1]. It was also necessary to install and configure the server [2], configure workstations and lots of other details.

An overview of our system can be seen in Figure 1. The Camsoft system consists of two main parts:

- Web User Interface – designed to convey information in an intuitive manner;
- Client – a program designed to carry out measurements on a given individual workshops.

The client is a fully-automated program that performs individual measurements driven by input data. The results of these measurements are converted into the desired form and output data are sent to the server, where they are further processed into final form.

The program acquires and processes information from the measurement of peripheral equipment (in this particular case, a mass measured using digital scales) and also captures the video of the entire measurement, which is obtained by security IP camera system. The video along with data from the measuring device are synchronized at the end of processing and sent to the server.

The web user interface is serving as a quick, easy, and visually acceptable interface to the data obtained from individual types of clients from various workplaces. The figures in the web user interface represent a video with time-synchronized data displayed during playback, including the following types:

- Data values – current value at the given time in video, such as effective weight.
- Complete record data obtained from measuring equipment – includes raw data, such as all measured weight.
- Main configuration information of clients – includes information like the time of turning on and off of the program, overview of the current file being sent to the server, etc.

---

* Mentor: Michal Kottman, Institute of Applied Informatics, Faculty of Informatics and Information Technologies STU in Bratislava

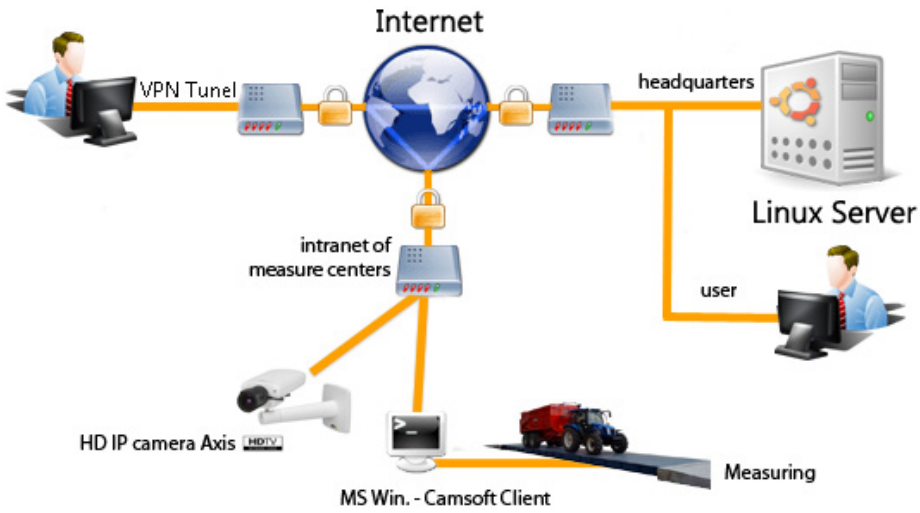


Figure 1. Overview of our Camsoft system for intelligent control.

The web user interface also includes features for remote configuration of all configuration data of individual clients. A further possibility of the interface is a centralized upload of an update file containing a new version of the client. After uploading the file, upgrades take place throughout the system automatically, updating all clients and their components to the latest version. After this update, the system automatically resumes normal operation.

In order to access the data, our web user interface offers the opportunity to create additional user accounts for viewing obtained information. It is possible to restrict access to individual functions of the interface for these accounts and set allow access only to the information necessary for the work of the user.

Our system is already being used in the company Achp Levice, a.s., which allowed us to implement our project. The system is in operation already for about 5 months. Due to the simplification of the entire control operation, the company experienced a significant decrease in losses from their income throughout all their workspaces.

At the moment we are handling formalities regarding our system, such as copyrights, through the legal channels, so that we would be able to distribute our system to spread further, since we heard from other companies dealing with measuring activities and they expressed serious interest.

In the future we plan update of system to link it with an unspecified economic system, which will result in an even better quality and control assurance.

Furthermore, we plan to advertise and distribute the system around Slovakia and maybe abroad later on.

References

[1] AXIS Communications. *AXIS Media Control*. [online]
Available at: http://www.axis.com/techsup/cam_servers/dev/activex.htm

[2] Ubuntu documentation team. *Ubuntu Server Guide*. [online]
Available at: https://help.ubuntu.com/12.04/serverguide/serverguide.pdf

The Analysis of the User's Behaviour

Marek BRIŠ*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
marek.bris@gmail.com

Extended Abstract

Our experiment is realized in XAPOS (experimental adaptive personalized ontology-based web system [1]), which belongs to the domain of e-learning web systems. XAPOS is a role-based system, for this text important roles are student and teacher.

There are many systems based on user behaviour such as *ALEF* system [2] which deals this topic as well. *ALEF* defines the modelling of the domain, extensible personalization, course adaptation and ultimately the active participation of a student in the learning process.

We consider the possibilities of motivation for active behaviour of users in role student in order to obtain as much relevant data as possible. We collect data in active way (activities performed by users in their passage through the content of the system) and in the passive way (information recorded without direct awareness of the user). Information retrieved from users in a passive way is – time spent on the site, – the number of visited pages, – sequence of visited pages, and in an active way – adding comments, – adding links to external sources, – evaluation of keywords, – taking tests.

We want to offer the obtained data to the user in the role student (student can see its own progress in XAPOS and achieved activities), and to the user in the role teacher. Teacher can evaluate the course flow and content of the course.

We performed an experiment in XAPOS over the content of *Programming in C-language* course, we motivate users to learn it and to collect bonus points for their activities, levels-of-game approach were chosen, as familiar environment accepted by the students. Acquired bonus points were counted to student's evaluation in the real final subject results. We categorize users according to their behaviour in XAPOS, with the aim to define the data relevance.

Users in XAPOS can participate in the following activities. *Adding comments* – the aim of comments is to obtain feedback on the quality of the system content. *Evaluation of keywords* – LO (Learning Object) is composed of concepts, which are described by the keywords. Since keywords have been generated by an external system, they may not always correspond to that concept and hence to the LO. So, we can improve ontological model. The user is given three possibilities to evaluate keyword (keyword is described correctly, description is absent, keyword is described incorrectly). *Adding external links* – the user can add to each LO a link (URI) to external sources (can be used as hints to students/users where learning material can be complemented). *Taking tests* – the user can take the test after each chapter, so he can verify the acquired knowledge.

---

* Master degree study programme in field: Information Systems
Supervisor: Assoc. Professor Petr Šaloun, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 403–404.

We suppose that students can take the opportunity to acquire bonus points differently:

- a group of hard-working students despite of sufficient amount of points from the subject can still aspire to acquire bonus points,
- next group can be "lazy" students, who are satisfy with comfortable but low amount of points from the subject and they will not go after bonus points,
- ordinary students spend some time in the system without extra interest,
- some students who already put a big effort in subject and have high amount of points, they cannot have an interest in bonus points,
- and finally, students who have a shortage of point in subject have to try to acquire as many bonus points as possible.

We verified this presumption in the following experiment. Over 250 partial-time and full-time students had taken part in *C programming language* course. We have had more than 16 000 records of evaluation of keywords, about 1 500 comments, 1 400 added links, and more than 2 200 taken tests.

Students who took part in the course could obtain from 0 to 10 points for their activity. We divided students into three groups – extra sedulous (over 8 points), – sedulous (over 3 points up to 8 points), – non sedulous (up to 3 points).

We had to modify the range of points in the subject, to compare them in the same scope with those ones in XAPOS. We used range from XAPOS (0-10 points). We calculated the difference between points in XAPOS and points in the subject for each student. The difference means how similar is behaviour (sedulity) between XAPOS and the subject (see Table 1).

Table 1. Similarity in users´ behaviour in XAPOS and in the subject.

Amount of students	*Difference in points*	*Behaviour in XAPOS vs. real subject*
40 %	2 points	Similar behaviour in system and subject
50 %	2 – 6 points	Higher difference in behaviour, not extreme
10 %	more than 6 points	Extreme differences in behaviour

Extension of XAPOS about activities allows users to add comments related to the quality of content, add third party links related to the content and more. Users can evaluate links of others users. XAPOS have not had this collaborative approach till now. We can affirm that users use the extended functionality actively for sharing information between themselves and for the feedback for the teacher (author of the content). So, users can share useful information, teacher can evaluate activities and the feedback, and the quality of system' content improves. All these three parties can profit from our extension of XAPOS.

In future works we want to analyze paths of users' movement in the system, improve the ontological model according to evaluation of keywords, and to offer to the teacher more options of the further use of data in an appropriate graphical interface.

References

[1] Šaloun, P., Velart, Z., Nekula, J.: Towards automated navigation over multilingual content. In: *Semantic Hyper/Multimedia Adaptation, Studies in Computational Intelligence*, Vol. 418/2013, Springer, 2013, pp. 203–229.

[2] Šimko, M., Barla, M., Bieliková, M.: ALEF: A Framework for Adaptive Web-Based Learning 2.0. In: *Proc. of IFIP Advances in Information and Communication Technology*, Vol. 324/2010, Springer, 2010, pp. 367–378.

User-Friendly Simulation of Wireless Networks in ns-3

Martin ČECHVALA, Ivana HUCKOVÁ, Jakub OBETKO,
Richard ROŠTECKÝ, Juraj ŠUBÍN, Viktor ŠULÁK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
tp@lists.kveri.com

Extended abstract

Wireless networks represent significant part of computer networks nowadays. Connection without the need of additional wiring is comfortable and widely used in schools, offices, public places or airports. Detailed proposal of every component of the network is crucial for implementing reliable and secure network. To verify correct functionality of the network simulation is used.

The goal of this project is to create a complex user-friendly framework for simulation of wireless networks. The core component of this framework is the ns-3 simulator, which is widely used and still in development. Because using the ns-3 simulator is a non-trivial complex task, which requires knowledge of Unix OS, C++ programming and knowledge of structure of ns-3 input and output files, our solution allows users without this knowledge to use the ns-3 simulator without necessity of learning them. The system is designed to be web-based, so that users do not have to install the ns-3 simulator or any other programs in order to simulate network function. The simulator runs on server connected to Internet and it is accessed by users using web framework.

The system contains tools for creating input for the ns-3 simulator, e.g. topology editor, which allows users to create network topology and set its parameters in GUI instead of writing a simulation script. There is also possibility of using Script Generator feature, which generates random ns-3 input script or to insert a script written by user. After the topology is prepared and parameters are set, user can start the simulation. The system will create ns-3 simulation script using information (topology and parameters) passed by user into the GUI or it will just use written script inserted by user. Script written by user will be checked during the compilation. If the compilation fails, user will be alerted. Script is then processed as an input with ns-3, which creates output files containing simulation results. Those results are in non-user-friendly format, and they have to be processed by other tools to create e.g. graphs. Our solution integrates some tools, which are able to transform simulation results to graphs. The user chooses which parameters he/she wants to follow during the simulation and after the simulation, relevant graphs are created. Tools used in our solution contain AWK program to parse output information or gnuplot to create graphs.

---

* Master degree study programme in field: Computer Engineering
Supervisor: Dr. Peter Magula, Institute of Computer Systems and Networks, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 405–406.

User interface of our project is meant to be simple and user-friendly, using drag-and-drop technique for creating desired topology and setting its parameters. It consists of two parts – graphical input generator (Figure 1) and library for user created topologies.

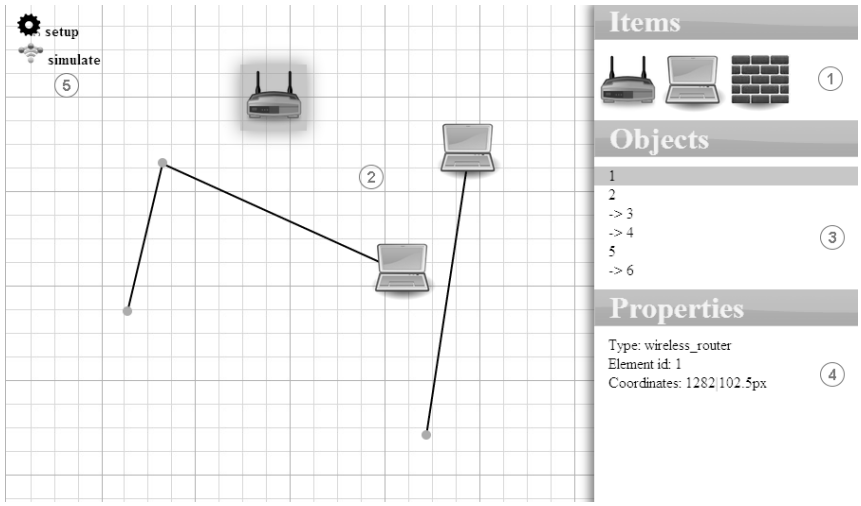


Figure 1. GUI screenshot – Graphical topology generator.

Our solution offers complex extension of ns-3, which combines most of the functionality of existing supporting tools for ns-3, e.g NS3Generator [1], Inet [2], TraceMetrics [3] or FlowMonitor [4]. And that is the major advantage of our solution – complexity, because each of examined tools is focused only on one aspect of work with ns-3, either input or output simplification. Our solution, on the other hand, provides complete user-friendly interface for inputs of simulation, as well as comfortable representation of results of the simulation. With its distribution as an open source, the proposed system will be available to wide range of end users, offering them an easier way of using the ns-3simulator. The opportunity of further development of this system is also offered by the open source distribution.

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic under the contract No. VEGA 1/0722/12.

References

[1] NS3Generator. *NS3Generator*. [Online; accessed March 16, 2013.]. Available at: http://www.nsnam.org/wiki/index.php/Ns3Generator

[2] Jin, Ch., Chen, Q., Jamin, S.: *Inet: Internet Topology Generator*. [Online; accessed March 16, 2013.]. Available at: http://topology.eecs.umich.edu/inet/inet-2.0.pdf

[3] TraceMetrics. *TraceMetrics – A trace file analyzer for Network Simulator*. [Online; accessed March 16, 2013.]. Available at: http://www.tracemetrics.net

Carneiro, G.: *FlowMonitor – a network monitoring framework for the Network Simulator 3 (NS-3)*. [Online; accessed March 16, 2013.]. Available at: http://paginas.fe.up.pt/~mricardo/doc/conferences/nstools2009/flowmon-paper.pdf

Promoting Educational Content by Use Cases

Matej ČERVEŇÁK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
matej.cervenak@gmail.com

Extended abstract

Lifelong learning is a continuous process of learning, in which individuals learn skills and knowledge of the field or profession. The contents of the educational area are not clearly defined [2]. Therefore, the content of education in the same areas can be different. It can easily happen, that, students learn knowledge useless in their field, and vice versa, they do not have access to useful one. It is caused by inaccuracies in the content in different languages and texts [2].

One of the possible solutions, how to eliminate the problems could be using information models for modeling educational content areas. In short, using software tools could represent curriculum, which is now noted in writing only. As seen on the models listed below, the advantage of this form of creating and writing the curriculum is more transparency and better comparability with others or same curricula. A comparison of various models can be important in determining the completeness and redundancy of educational content. On such comparisons, it is possible to determine changes in individual contents. Based on sightings, educational institutions can update their curricula and thus contribute to the improvement of learning processes. On the other hand, students, having access to the models, would be able to better deal with the decision concerning their future.

Choice of the curriculum is identified by objectives of teaching process, which is the exact idea of what is to be achieved. Objective in the area of information systems are often referred to as the output step. Education has not done the necessary steps required to the defined learning objectives precisely, clearly and systematically defined. Therefore, this area remains incomplete in the field of education. Many of students leave educational institutions with the knowledge, skills and abilities that cannot take advantage on the labor market. Therefore, it is necessary to develop a system of education developing the skills and abilities of students. [1, 2]

Events are important in terms of educational content. We argue the events determine the content of education. In this case, the events are specific questions and we look for their answers. Questions can be requirements to student or employee, while the answers to the questions are educational content after correct formulation.

Another important concept in terms of educational content is socially acceptable level of education. This is the minimum level of education, which recognizes the company. If we consider the minimum content of education, we can socially acceptable level of education and minimum level of education consider as equivalent. This implies, that the minimum content of education deduces and determines us the required level of education.

---

* Master degree study programme in field: Information Systems
Supervisor: Dr. Ján Lang, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 407–408.

In Slovakia, each job position offered by employment offices has its own description (`http://www.istp.sk/ktp`). This description of type position (working operations and conditions) or particular profession has big disadvantages. Those are ambiguity and incompleteness. Ambiguity is evident in the comparison various the descriptions. Another issue that is closely linked to the mentioned ambiguity is incompleteness. It is common, that in order to work on a specific position we expect something else, than expect our employee.

Based on the above mentioned problems with the description of the type positions we can argue that it becomes useless. Software tools can get space right here to eliminate these problems. For the modeling of the curriculum is used *RUP* methodology with *UML* support, specifically use cases created in software tool *Enterprise Architect*. As well as it is possible to create educational content and curriculum for school subjects, it is possible to identify such content for work professions. In the following example are created use case diagram, which shows the connection between the type position of *Payroll accountant* and school subject *Mathematics*. In modeling the knowledge and content of education in diagram are used *include and generalize* relationship, which enables us to capture the relation between use cases.

In the *Figure 1* is displayed use case diagram of type position *Payroll accountant*. As an actor in the model perform employer. He requires from an employee, represented by information system, knowledge of individual use cases. The diagram also shows the model of the school subject *Mathematics*. Diagram covers part of content of primary and secondary school. If we imagine, that the use cases in the diagram of school subject are necessary knowledge of secondary school student, we can consider content of the subject as socially acceptable level of education. These two models are linked by *include* relations. We can see, which knowledge of the school study is required from executor of profession. It is also seen, that executor of profession do not use the use case *ComputeFactorial* in his work, even it is part of the secondary school leaving examination and the knowledge is required of him.

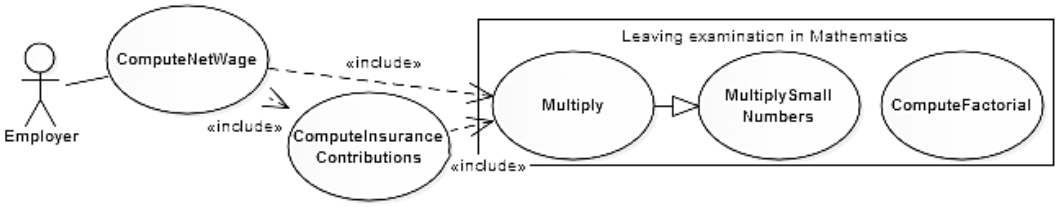


Figure 1. Use cases of employee Payroll accountant and school subject Mathematics.

The method of creating educational content with use cases gives us ways to avoid the current problems. Content can be easily compared and checked on the level of redundancy and relationship between different contents. This approach can be used for further research in the field of using *UML* as a tool for the creation of educational content. Together with other *UML* diagrams can be substituted for the current verbal description of educational content.

Acknowledgement. This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/1221/12.

References

[1] Mylopoulos, J.: Conceptual Modeling in the Time of the Revolution: Part II. Gramado, Brazil, 2009.

[2] Turek, I.: Ciele vyučovacieho procesu: Kapitoly z didaktiky, 1. vyd. Bratislava : Metodické centrum v Bratislave, 1995. 48 s. 80-85185-93-8. (in Slovak)

Emotion-Aware Movie Recommender Based on Genre Impact Analysis

Dominika ČERVEŇOVÁ*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
cervenova.dominika@gmail.com

Extended Abstract

This abstract contains a proposal of a context-aware knowledge-based method that recommends movies based on users' current emotions. Recently users' mood has shown up as an important context feature, relevant for recommending systems. It has became an object of interest for many researchers. A work of Shi et.al [2] or Wang et.al [3] or an interactive web radio, called Musicovery (musicovery.com) represent some of many works, where authors take users' mood into account.

Our method is based on assumption that there is a relationship between users' current mood and movie genre suitable for her at the specific moment. With the knowledge of how exactly specific genre influences emotions and provided that we have the information about users' current mood, we are able to determine which genres are the most suitable and make the decision which movie to watch much easier for user.

The method uses postfiltering of data from a metadata-based recommendation service, provided by project called TeleVido (team01-12.ucebne.fiit.stuba.sk/web). This service recommends user a list of movies, that might be interesting for her in general. However we try to identify what user might find interesting at the moment, to make the recommendation even more personalized and this is where the emotions help us. After getting the resulting list of recommended movies from the service, we take an information about users' current mood, she gives us explicitly. Then based on defined binding rules between genre and mood, the method reorders the list of movies into a new list (eventually eliminates some items, potentially irrelevant according to the gained mood). A schema of our recommendation method can be seen in the Figure 1 below.

To acquire the rules for recommendation we used several approaches. We started with some simple rules based on our opinions, consulted the movie-mood relationship with a psychologist and interviewed 10 randomly selected web users, various age and interests. As the second step of rules extraction we tried to mine some association rules from the LDOS-CoMoDa dataset [1]. This dataset contains users and movies with many contextual information (including users' feelings before during and after watching particular movie). These can help us to find mood influence of genres on emotions and can be used for experiments to evaluate recommendations created by our method as well. The

* Bachelor study programme in field: Informatics
Supervisor: Dušan Zeleník, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

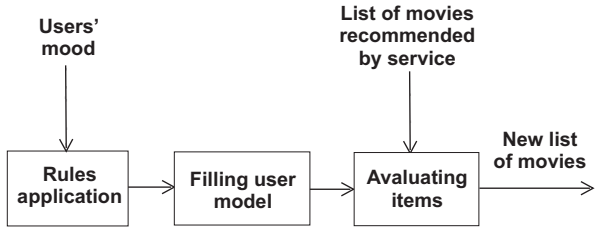


Figure 1. Schema of the recommendation method functionality.

result of this process was a table of percentage occurrence of particular genre in positive, negative and neutral mood. From this table we found out, there are some genres basically independent on users' emotions (e.g. Crime), but in many cases there were observable differences between frequency of choices in positive, neutral and negative mood. For example Drama appeared to be more wanted by negatively tuned people. On the other hand Comedy is preferred by people with positive mood and an occurrence of Adventure was twice lower by negative mood than in positive and neutral case. We created rules that bind particular genre to particular mood. The rules are represented by binding matrix, where $value[i; j]$ represents desirability of $genre[i]$ in $mood[j]$.

Our user model contains a desirability or relevancy of each genre in the context of current mood represented by a value computed according to binding values. The user model is filled every time we get a new information from a user about his emotions. After the user model is filled, we evaluate each movie from the list recommended by TeleVido by calculating an average value from desirabilities of genres specified for the movie (Equation 1).

$$value = \frac{d_a + d_b + d_c + dots}{genre\_count} \tag{1}$$

In the end, the movies are sorted descending into a new list and the items with value grater than $-0,8$ are shown to user as the most proper recommendations. The evaluation focuses on reordering the list of recommended movies by giving them higher or lower ratings, accordingly to relevance of movie genres in user model at the moment. The aim is also decreasing a number of recommended items, if possible, by skipping some of them with the lowest ratings.

Our recommendation method is currently being implemented and we already made some experiments with explicitly acquired context, using the LDOS-CoMoDa dataset that proved our hypothesis. In addition we are about to make some qualitative experiments with real users. A comparison between items recommended without our method and the resulting list after postfiltering applied and also a following feedback from users can fully confirm the relevancy of our method.

Acknowledgement: This work was partially supported by the Slovak Research and Development Agency under the contract No. APVV-0208-10.

References

[1] Košir, A., Odić, A., Kunaver, M., Tkalčič, M., Tasič, J.F.: Database for contextual personalization. In: *Elektrotechiski Vestnik 78(5)*, 2011, pp. 270–274.

[2] Shi, Y., Larson, M., Hanjalic, A.: Mining mood-specific movie similarity with matrix factorization for context-aware recommendation. In: *Proc. of the Workshop on Context-Aware Movie Recommendation - CAMRa'10*, 2010, pp. 34–40.

[3] Wang, L., Meng, X., Zhang, Y., Shi, Y.: New approaches to mood-based hybrid collaborative filtering. In: *Proc. of the Workshop on Context-Aware Movie Recommendation - CAMRa '10*, 2010.

An Approach to Triple Based User Activities Logging and Classification

Igor DANIŠ*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
igordanis@yahoo.com

Extended abstract

The paper is focused to evaluate user interface friendliness, accuracy and smoothness. Lack of these characteristics can result in unsuitability for work and user loss. We propose method of triple based user activities logging and classification, in order to employ many great benefits of the Semantic Web technologies such as reasoning. Various user activities create RDF graph, which can be queried and processed with SPARQL or business rules.

The main goal of classification is to reveal problematic places in a web pages analysing if user action is desired or not. If an application provides inconvenient user interface there is high probability that application lose its visitors even though it has very advanced back end. Finding problematic places in application is very important and it should lead to taking corrective actions. Almost every application creates logs for further analysis today; therefore mentioned problem of finding problematic places should be accomplishable. Since Semantic Web technologies such as RDF [1], OWL [2] open new possibilities to work with data, we propose an approach to triple based user activities logging and classification with these technologies extended with reasoning capabilities of native triple stores.

Our motivation is to identify places in real estate web application which we consider critical because they breach positive user experience. Moreover, we need dynamically customize application or even reconsider use cases if it is found that users acts different than it was intended in process of analysis. With these results we have focused on user classification. In order to reach our goals, we needed to implement semantic logging approach. User session logs are stored in RDF graph form. Every user activity is stored as the RDF statement, i.e. triple in form subject – predicate – object. When the user executes first action in our application, the first necessary thing is to create statements about the session. Every user activity then can be linked with session. Log information is provided by implicit feedback running in asynchronous mode in background without need for user intervention. Application layer logging is supplemented by server side logging which maps session-related information to user. Logs recorded by application layer in co-operation with reasoning and ontology provide huge flexibility of data which we can log in terms of quantity and heterogeneity. With implicit logging we log events such as click on estate thumbnail icon, change of application localization, estates comparison etc. Collected data can be queried by means of SPARQL in order to extract knowledge about different aspects of application.

---

* Bachelor degree study programme in field: Informatics
Supervisor: Dr. Miroslav Líška, Datalan, a.s., Bratislava

Such an extracted aspect can be recognition of critical place. The following image depicts user logs registered during one of user sessions represented as statements in RDF graph form.

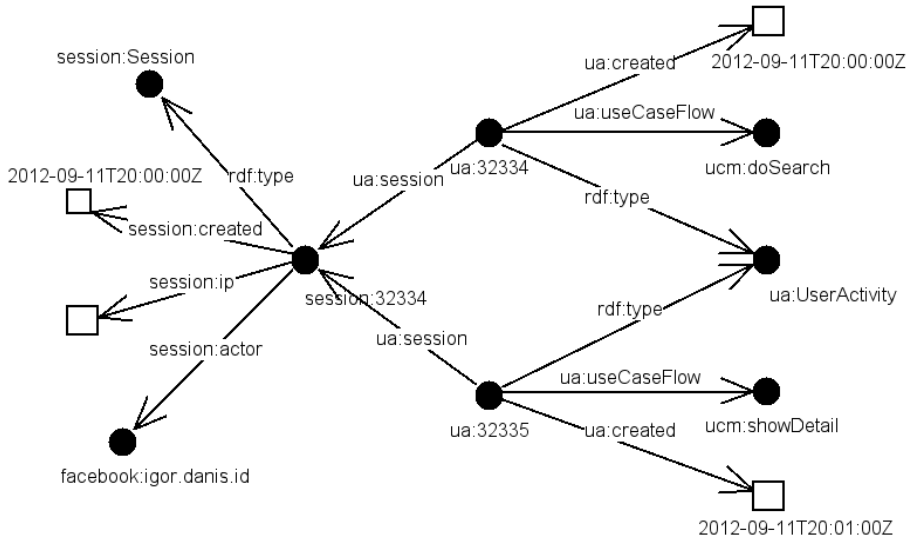


Figure 1. RDF Graph of user activities within session.

Every RDF statement was processed by rules given to reasoner which is provided by triple store database. This way we categorize users into groups by level of their satisfaction. Such rule is to evaluate user during search use case when he marks estate as his favourite as satisfied user. Similarly we evaluated users who were leaving application as unsatisfied. Beside typical user activities that were parts of intended use cases we discovered that considerable number of unsatisfied users was leaving application during visit of one specific web page. There was a bug in it which was not covered by unit tests hindering meaningful estate comparison. In future there is need for consideration of number possible improvements such as:

- filter users in reasoned groups depending on their similarity to other users,
- track reasoned results of satisfaction and confront it with explicit feedback provided by users.

These information will provide solid ground for improvements in further recommendation process.

References

[1] W3C. Resource Description Framework (RDF):Concepts and Abstract Syntax, (2004), [Online] [Accessed: February 20, 2013], `http://www.w3.org/TR/rdf-concepts/`

[2] W3C. OWL Web Ontology Language Guide, (2004), [Online] [Accessed: February 22, 2013], `http://www.w3.org/TR/owl-guide/`

[3] Bieliková, M., Barla, M., Tvarožek, M.: Rule-based User Characteristics Acquisition from Logs with Semantics for Personalized Web-Based Systems, *Computing and Informatics*, (2009), vol. 28, no. 4, pp. 399–427.

[4] Bieliková, M., Holub, M.: An Inquiry into the Utilization of Behaviour of Users in Personalized Web, *Journal of Universal Computer Science*, (2011), vol. 17, no.13, pp. 1830–1853.

Identifying Relationships among Entities of Digital Libraries Revealing the Originality of Sources

Zoltán HARSÁNYI*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
harsanyi@fiit.stuba.sk

Extended Abstract

Digital libraries were viewed as systems providing a community of users with coherent access to a large, organized repository of information and knowledge. Research in digital libraries in the past years has resulted in a wide range of technological development. However, digital libraries require research in many other areas including dynamic interoperability, support for library evolution, contextual search mechanisms, and social issues as proposed in many past studies [2]. In this paper, we focus on*semantic interoperability*. The variety of metadata standards, the existence of local schemes and different ways of metadata usage and their implementations have significant implications for individual institutions to provide access to its information resources and try to share its content and metadata among other DLs [1].

The most common way of achieving semantic interoperability in DL (independently from the technical aspects like transmission protocols, shared databases, etc.) is using conceptual reference models, i.e. high level ontologies. Ontologies and metadata provide the specific tools to organize and provide a useful description of heterogeneous content [3]. Developing specific ontologies can enable interlinking context with the objects itself, why machines can also capture the semantic of these relationships. The concept Linked Data or Web of Data presents the practical usage of ontologies for interlinking objects of different types from different domains. Presenting information based on the Linked Data recommendations moreover enhances the machine readability of these data. Hence we focus in our further research on these approaches.

The area, we are focusing on are bibliographic databases. These databases contain highly structured records of scientific publications from different research domains. Since the records are stored in a structured way, identifying the semantics of the data does not present a huge issue for different systems. These libraries contain a huge number of publications. Let's imagine a situation, when a young scientist would like to make a research in a chosen domain (e.g. digital libraries, semantic interoperability, social engineering, etc.). The first task he needs to do is collecting a big

* Doctoral study programme in field: Program Systems
Supervisor: Assoc. Professor Viera Rozinajová, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 413–414.

number of papers dealing with his research area. He focuses on actual publications to get the "actual" state of research. What if the studied area is "popular" enough for other researchers and the most of the publications concerns with specific issues on the are. The ideal solution for him would be starting from the "ground" and trying get in the area first. But how to achieve this? How to find the "source" publications? How to find out the evolution of the area? How to identify the average contributions of the successive publications? These questions could be summarized as *how to model the given research area*? Under modelling a research area we mean some form of abstract representation of the area with "arrows" to and from the related areas with some kind of contribution index. This kind of model could be used to navigate through the whole area to find out, which research areas are the most popular, how they evolved, etc.

Several ontologies exist with the aim of capturing the relationships in bibliographic databases, e.g. BIBO, VIVO, FOAF. These ontologies are formalizing the basic relationships, which can occur in these databases. BIBO provides main concepts and properties for describing citations and bibliographic references (i.e. quotes, books, articles, etc.) on the Semantic Web, FOAF is a project devoted to linking people and information using the Web, VIVO enables the discovery of researchers across institutions. None of the existing solutions are capable enough to model a complex domain with the aim of detecting research areas, finding source publications in this area.

By extending and interlinking the existing ontologies, the proposed model could be created to visualize the evolution of the research areas. By analyzing the existing bibliographic databases and putting their content into Semantic Web using the Linked Data recommendations, these relationships could be processed also by other systems, so the ontology can be used to enhance the semantic interoperability between different systems. Our aim is to apply the designed ontology to different databases and analyze a research area not only from one source, but from multiple publishers. This method can also lead to find which publishers can be considered as the most significant.

To design a model described above, we need to perform the following tasks: analyze the existing ontologies concerning with bibliographic databases (publications, citations, institutions, people, etc.), design an ontology based on the patterns while focusing on the capturing of the evolution of a research area, create the ontology applying Linked Data recommendations, apply the ontology on existing libraries to create the model. By achieving the above mentioned goals, the contribution of the method could be summarized into the following points: allowing the modelling of a research area, the model can be used to reflect and visualize the evolution of a research area, the model can be applied to many different sources the get the best possible state of the area, the analyzed areas can be shared among other systems and it can boost the users experience while travelling in the digital space. Currently our research dealing with this specific issue is in early stages, so we have just begun the implementation of our methods but we are not so far from real experiments. We divided the design of our method into the following periods: designing an ontology capturing a research area (ongoing), implementing the ontology using OWL, comparing and evaluating the designed ontology against the existing and mentioned ones above using of ontology matching.

Acknowledgement: This work was partially supported by the Slovak Research and Development Agency under the contract No. APVV-0208-10.

References

[1] Freund, L., Rasmussen, E., Sugiyama, K., eds.: Semantic Metadata Interoperability in Digital Libraries: A Constructivist Grounded Theory Approach, 2011, ACM/IEEE Joint Conference on Digital Libraries, Ottawa (Canada), 13 June 2011.

[2] Ram, S., Park, J., Lee, D.: Digital Libraries for the Next Millennium: Challenges and Research Directions. *Information Systems Frontiers*, 1999, vol. 1, no. 1, pp. 75–94.

[3] Staab, S., Studer, R., eds.: *Handbook on Ontologies*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

Modeling the Domain of Software Development to Represent Skills of Programmers

Michal HOLUB*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
holub@fiit.stuba.sk

Extended abstract

Many datasets published on the Web use Linked Data principles [2]. Such a dataset is composed of concepts and links to describe entities from a certain domain and relationships between them. We can also use it to represent the knowledge or skills of people.

Linked Data are being used in specialized datasets (describing a concrete domain) as well as in general datasets capturing the general knowledge about various entities. A rising number of datasets is being extracted from free online encyclopedia Wikipedia [4].

The datasets using Linked Data form a Linked Data Cloud. In the center of this cloud there are two large datasets: DBpedia [1] and YAGO [3]. Both use Wikipedia as their primary source of information, they extract it from infoboxes and categories. These datasets define as many entities as possible, so that other datasets can link to them.

We would like to have the knowledge of software developers represented in such a way that enables us to search it and navigate between people efficiently. This would allow us to get an overview of capabilities of a group of people, to compare them against each other, to search for a person suitable for a particular task, search for colleagues in order to help with a problem, etc.

We propose a domain model describing the area of software development using concepts representing entities from this area. We also propose a method for automatic population of this ontology. The main feature of the proposed method is matching the concrete technologies (instances of concepts) with their types, i.e. determining the class for each particular technology (principle, protocol, etc.). To the best of our knowledge, no similar ontology exists.

Concepts are linked together using relationships like *is part of*, *is a*, *is written in*, or *uses*, as shown in Figure 1. Thanks to the relationships we can later do reasoning, e.g. deduce that when a programmer knows jUnit (a testing framework for Java) he also has to know a bit of Java (a programming language), because there is a relationship stating "jUnit uses Java".

As a source of information for the concept map population we use free online encyclopedia Wikipedia. We analyze the textual content of its articles to learn new concepts and their instances.

We use the existing concepts as a seed in the task of ontology learning. We search all Wikipedia's articles for occurrences of concepts from our concept map. Take "programming language" as an example of a concept. Article about it also links to a "List of programming

---

* Doctoral degree study programme in field: Software Engineering
Supervisor: Professor Mária Bieliková, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

languages", from which we can extract additional concepts, which are subclasses of "programming language".

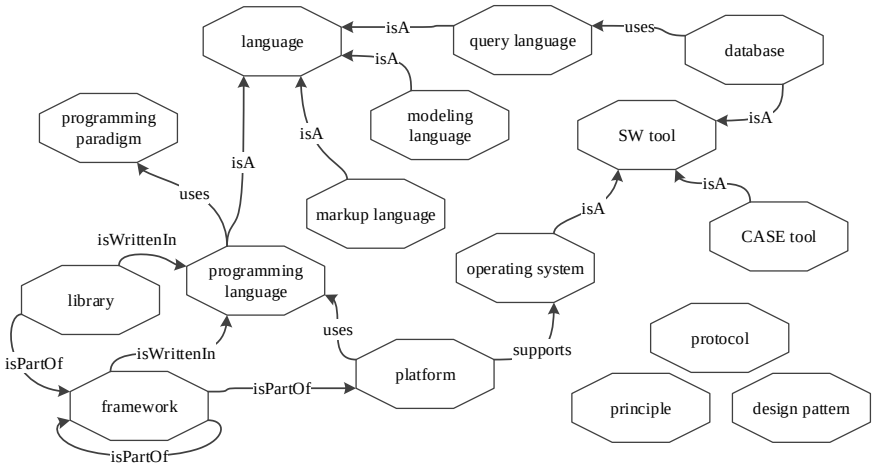


Figure 1. Concept map for the domain of software development.

When the ontology is ready, we populate it with instances, which we do as follows:

1. Select an article from Wikipedia containing a particular concept in its text.
2. Find the first sentence containing the verb *is* followed by one of the concept types.
3. Convert the title to a new concept instance (if it is not present) and create *is a* relationship between the instance and the concept.

Using this process we not only populate our domain model with particular technology names, we also find all terms which can describe a technology used when developing software.

There can be other words following the verb *is* in the article not matching any concept from our map. These could express properties of the technology and we might enhance the ontology.

The ontology can be used in a system for gathering the knowledge of programmers. Let us assume the user adds "Java EE" to his skills. We identify it as a platform in the ontology. It is related to "programming language" by *uses* link. We can generate question "Which programming language used in Java EE are you familiar with?" This way we can get more skills from the user.

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/0675/11.

References

[1] Auer S., Lehmann J.: What Have Innsbruck and Leipzig in Common? Extracting Semantics from Wiki Content. In: *The Semantic Web: Research and Applications*, LNCS Vol. 4519. Springer, (2007), pp. 503–517.

[2] Berners-Lee, T.: *Design Issues: Linked Data*. [Online; accessed March 19, 2013]. Available at: http://www.w3.org/DesignIssues/LinkedData

[3] Suchanek F.M., Kasneci G., Weikum G. YAGO: A Core of Semantic Knowledge Unifying WordNet and Wikipedia. In: *Proc. of the 16th int. Conf. on World Wide Web*, ACM Press, (2007), pp. 697–706.

[4] Wong, W., Liu, W., Bennamoun, M.: Ontology Learning from Text: A Look Back and in the Future. *ACM Computing Surveys*, (2012), vol. 44, no. 4, 36 p.

Augmenting Web for Facilitating Learning

Róbert HORVÁTH*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
roberthorvath89@gmail.com

Extended abstract

We spend large amount of time browsing the Web and we come across a lot of documents. We believe that this amount of time can be spent more effectively due to integrating text augmentation methods into web browsing. Providing user with additional information can help in education process, for example foreign language learning.

Main goal of our work is to create a method for web augmentation for facilitating foreign language learning. We enrich web content by replacing appropriate words during web browsing, maintain user knowledge and user preferences, while considering specifics of learning process such as forgetting. The method brings together process of web browsing and vocabulary learning. Potential for this approach is supported by advances in technology-enhanced learning and computer assisted language learning. It was shown that learning occurs even unintentionally and with minimal mental processing [1].

There already exist approaches to enhancing webpages to help user with learning foreign language unintentionally. Most of them are implemented as web browsers extensions. Analysis shows that they avoid user knowledge modelling, which leads to random presentation of foreign vocabulary [2, 3]. In contract to them, Duolingo provides learning platform based on user model and approach which considers specifics of learning process. Studies show that its effect on language learning is comparable to school classes [4].

Our method for webpage content augmentation provides user with opportunities for vocabulary learning without intention of studying. Our aim is to find appropriate terms for learning in webpage content user is going to read and replace them with their translations the way user is still able to understand meaning of content and remember the vocabulary. Our method consists of three main steps executed for every visited webpage (see Figure 1):

1. *Text analysis and pre-processing* – Unnecessary information is removed, webpage is translated and translations are mapped to user vocabulary knowledge to find the best candidates for learning.
2. *Personalized text augmentation* – We replace and highlight words on webpage to present new vocabulary, while preserving original meaning.
3. *User model update based on user activity monitoring* – User activity is monitored and based on his/her behavior (time spent on webpage, interaction with text, etc.) user knowledge model is updated.

---

* Master degree study programme in field: Software Engineering
Supervisor: Dr. Marián Šimko, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

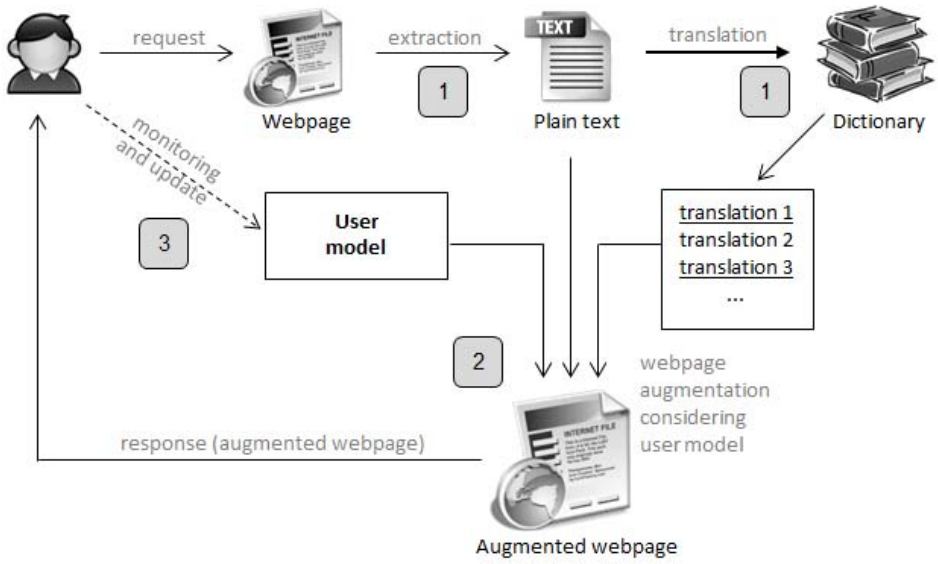


Figure 1. Process of personalized webpage augmentation.

In order to evaluate our method we have created web browser extension for Google Chrome. It successfully utilizes our approach and it is able to augment Slovak webpages with English vocabulary, which is derived from user knowledge model. For evaluation purposes extension gather both implicit feedback from monitoring user activity and explicit feedback from regular vocabulary tests. To find the effect on the learning process we propose two main hypotheses:

1. Augmentation improves foreign language vocabulary size.
2. Time spent with reading augmented webpages will increase insignificantly.

We have already conducted small supervised experiment to evaluate effect of text augmentation of reading speed. The results show that augmented webpage slows reading speed down on average by approximately 7%. We find these results very reasonable with a great potential to support the second hypothesis. However, we need to conduct further experiments using larger data set to obtain more significant results.

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/0971/11.

References

[1] Groot, P. (2000). Computer Assisted Second Language Vocabulary Acquisition. *Language Learning & Technology*, pp. 60–81.

[2] Streiter, O. et al. (2005). Browsers for autonomous and contextualized language learning: tools and theories. *Information Technology: Research and Education, 2005. ITRE 2005. 3rd International Conference on , vol., no.*, pp. 343–347.

[3] Trusty, A., Truong, K. N. (2011). Augmenting the Web for Second Language Vocabulary Learning, *Proceeding of the 2011 annual conference on Human factors in computing systems*, ACM New York, USA, pp. 3179–3188.

[4] Vesselinov, R. (2012). Duolingo Effectiveness Study. City University of New York, USA.

Reduce the Power Consumption by Selecting the Appropriate Processor

Peter JOMBÍK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
jombik@fiit.stuba.sk

Extended abstract

Processors performance in mobile devices increases in last years. Current processors used in mobile devices (e.g., mobile phones, tablets etc.) have computing power like older desktop computers. In 2000, average phone battery life was 1 week. Current phones have greater computing power and more peripherals, but with greater power consumption. Two architectures are used in processors – RISC and CISC. These architectures were used in different devices until last years. The ARM architecture describes a family of RISC-based computer processors designed and licensed by British company ARM Holdings. It was first developed in the 1980s and globally as of 2013 is the most widely used 32-bit instruction set architecture in terms of quantity produced. Intel Atom is the brand name for a line of ultra-low-voltage IA-32 and x86-64 CPUs (CISC) from Intel. Developments and changes in the characteristics of these two different processors allow the use of similar devices.

To compare processor we must choose right operating system. In Table 1 are showed basic OS for both processors based on ARM and x86 architecture.

Table 1. Support for operating systems in ARM and X86 architectures.

ARM	x86
Windows CE, Windows RT	Windows CE, XP , Vista, 7, 8
Android, Linux, Chrome OS	Android, Linux, Chrome OS

First test was based on maximum performance of processors. Frequency of processors was lowered on same value 1 GHz in second test. Ubuntu 12.04 was used. Results are showed in Tab. 2.

Table 2. Maximum performance and same frequency test.

Test bench 1	OMAP4460	Samsung NC10
CacheBench (Read/Modify/Write)	2449.28 MB/s (better)	1901.24 MB/s
LAME MP3 Encoding (Wav to MP3)	107.28 s (better)	137.04 s
FFmpeg (AVI to NTSC VCD)	199.70 s	76.38 s (better)
GraphicsMagick (Resizing)	18 iterations/m (better)	14 iterations/m
NAS Parallel Benchmarks (BT.A)	411.02 (better)	413.02
X264 (H.264 Video encoding)	4.04 frames/s	4.92 frames/s (better)

* Doctoral degree study programme in field: Applied Informatics
Supervisor: Assoc. Professor Tibor Krajčovič, Institute of Computer Systems and Networks, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 419–420.

Test bench 2	Cortex-A9 1GHz	Intel Atom N450 1GHz
LZMA Compression (256MB file)	797.02 s (better)	942.81 s
LAME MP3 Encoding (Wav to MP3)	122.64 s (better)	163.11 s
Dcraw (RAW to PPM Image conversion)	694.64 s	615.20 s (better)
GraphicsMagick (Resizing)	17 (better)	7

Compare two computers with similar HW specifications are necessary for power consumption. In table 3 are result of compare two computers with same parts but different processors.

Table 3. Power consumption.

Display off	Chromebook 303 Cortex-A15	Chromebook 500 Atom N570
Idle	4.07 W	8,12 W
Kraken (avg)	8.32 W	11,4 W
Kraken (peak)	9.27 W	12.4 W

We created evaluation table 4 for processors Cortex and Atom. We can choose the recommended processor for selected application using this table.

Table 4. Evaluation table.

Type of processor:	Cortex-A	Intel Atom
Basic number of points:	100	100
Frequency in GHz:	*1	*0,8
Number of cores:	*0,5	*1
Number of threads:	*1	*0,5
Advanced multipliers:	*1	*0,5
Have integrated graphics? :	*1	*1,1
Contain NEON technology? :	*1,2	N/A
Contain Jazelle technology? :	*1,1	N/A
Contain SIMD technology? :	*1,1	N/A
Multipliers based on deployment:	*1,1	N/A
Power consumption aware:	*1	*0,5
Picture or sound rendering:	*1,2	*1
Web application:	*1,2	*1
Use different operating systems:	*1	*2
3D application:	*1,3	*1

By using this table we can choose better processor for selected applications. This table is suitable for developers, software engineers or companies. They can select better solution for their project by set basic parameters of needs. This table can be used for devices like notebooks, tablets, servers.

References

[1] Hectronic: *ARM versus X86 – Considerations for the embedded segment* [Online; Accessed 21. 8. 2012], Available: http://www.hectronic.se/website1/embedded/arm-versus-x86/arm-versus-x86.php

[2] M. Larabel: *The ARM Cortex-A9 Can Beat Out The Intel Atom* [Online; Accessed 25. 10. 2012], Available: http://www.phoronix.com/scan.php?page=article&item=gentoo_arm_x32&num=1

[3] A. Lal Shimpi: *Samsung Chromebook (XE303) Review: Testing ARM's Cortex A15*, [Online; Accessed 5. 11. 2012], Available: http://www.anandtech.com/show/6422/samsung-chromebook-xe303-review-testing-arms-cortex-a15/7

Intelligent RSS Reader and Article Recommendation System

Richard KAKAŠ*

High School Jur Hronec
Novohradská 3, 821 09 Bratislava, Slovakia
richard.kakas@gmail.com

Extended Abstract

Nowadays there are millions web pages accessible through the Internet. Most of these pages provide text content to the visitors. This text is mostly in the form of articles which are often accessible via web syndication systems like RSS or Atom and their best effort to provide relevant information [4]. These systems make it easier for Internet users to track web sites content changes but they do not fully solve the major problem – filtering bunch of articles to show only those that might be interesting to particular user [2]. Such filtering can be done by recommendation systems. These systems are well-known across professionals and are available on many sites. They can be found on many e-shops, portals and also blogs where they are suggesting site's content to visitors.

Our web service is merging the benefits of web syndication and recommendation system. This service is downloading articles from RSS/Atom feeds, list of feeds is created and updated by us (of course user can send suggestions). Feeds for many pages and many article categories will be added over time. Therefore, everyone will find his topic on our web site without the need to create own list of RSS/Atom feeds. Links to all fetched articles are then visible on our web page. User interface allows article sorting by date, number of reads, rank and recommendation. It also provides filtering by feed domain, time range and article category.

Our recommendation system uses two approaches – collaborative and content-based filtering. Both can provide reliable recommendations that reflect the user's past behavior, however each is taking into account different data. Collaborative filtering does not analyze article's text but use just relations between articles and their readers to suggest appropriate article to the user [3]. On the other hand content-based filtering is designed to pick up similar article [1]. Similarity is determined by article's content.

By using both kinds of filtering we overcome unwanted cold-start of pure collaborative approach. Moreover, we are expecting better results than we would have by using non-hybrid filtering. Our intention is to implement both filtering approaches as single algorithm which operates over one graph that contains all data – relations between articles, readers and content data of every article. Data are stored in `Neo4j` database which is optimized for storing and processing huge graphs. Thanks to this we are expecting real time execution of the algorithm (see Figure 1). Our intention is to connect

* Mentor: Dušan Zeleník, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

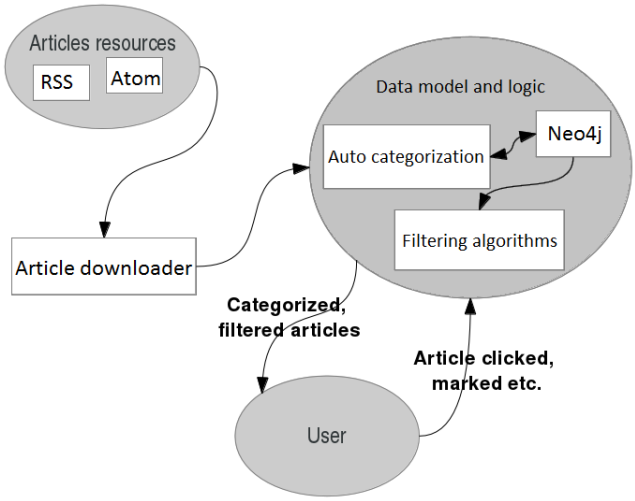


Figure 1. Overview of the method for RSS recommending and its logical flow.

articles using users who are interested in articles and keywords which are extracted from articles using TF-IDF.

Another benefit that our web service provides for the user is nicely categorized articles. This categorization should be done automatically [2]. However, to make auto-categorization possible we need a relatively huge amount of categorized content data. These data are stored in the same graph alongside with articles and readers and very similar algorithm to the one mentioned above will do auto-categorization. Categorized content data might be retrieved from category-exclusive resources which have to be added to our system from the beginning.

The web page itself is being programmed in `Java` using the popular web framework `Spring`. The recommendation algorithm resides in separate libraries to allow its easy reuse in other projects.

In future we might provide access to our algorithm and database through the API for web page/blog owners. With the API they might fill our database with their content without need of RSS and also they might embed our article recommendations on their own web site (of course only articles from their site will be visible). The collected data about users might be also used for ad targeting in the future.

References

[1] Bielikova, M., Kompan, M., Zelenik, D.: Effective hierarchical vector-based news representation for personalized recommendation. *Computer Science and Information Systems*, 2012, vol. 9, no. 1, pp. 303–322.

[2] Creus, J., Amann, B., Travers, N., Vodislav, D.: RoSeS: a continuous query processor for large-scale RSS filtering and aggregation. In: *Proc. of the 20th ACM int. conf. on Information and knowledge management*. ACM, New York, NY, USA, 2011, pp. 2549–2552.

[3] Das, A.S., Datar, M., Garg, A., Rajaram, S.: Google news personalization. ACM Press, New York, New York, USA, 2007, p. 271.

[4] Horincar, R., Amann, B., Artières, T.: Best-Effort Refresh Strategies for Content-Based RSS Feed Aggregation. In: *Web Information Systems Engineering*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2010, pp. 262–270.

Creating and Recognizing Visual Words Using Sparse Distributed Memory

Ján KVAK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
kvak@fiit.stuba.com

Extended abstract

The image recognition and classification is becoming an area of huge interest lately. However the reliable and fast algorithm that could recognize any object in input scene as fast humans are able to do, is still something to be discovered. For now the algorithms are always trade-offs between reliability, speed and number of classes that can be recognized.

There is an approach treating visual features as words and grouping these features to recognize objects called bags of visual words. The usual approach to creating and using visual words is to extract local descriptors from a set of input images divided to N classes. Then they clustered this set of local descriptors using K-means algorithm. [1] Clustering provided them with "codebook" of words, in which the words were centers of clusters. Then each image can be represented as a "bag of visual words". These bags of words can be trained into classifier.

In previous works concerning visual words, SIFT local descriptor was used to describe keypoints. In this work, we will use the latest one FREAK which is binary feature descriptor. [2] We can use the fact, that all local features are represented as binary vectors. Clustering and classifying now takes place in Hamming space, because we can compare these features using Hamming distance. For this purpose we propose the use of Sparse Distributed Memory (SDM) augmented with genetic algorithms also called genetic memory. [3]

SDM takes advantage of sparse distribution of input data in high-dimensional binary address space. SDM is an associative memory, which purpose is to store data, and retrieve them, if address we call is sufficiently close to the address, at which data were stored, then it should return data with less noise, than the noise in the original address.

Sparse Distributed Memory consists mainly from two parts – location addresses and data counters. It has constant radius that describes maximum distance to location address, in which this address is still selected. Then, we have reference addresses, which denote the classes, we want to train. The fact worth noting is that the number of reference addresses is much bigger than the number of location addresses. During the process of training we choose location addresses closer than radius to the reference, we want to store the data in, and store the data in data counters.

The problem is how to choose location addresses, to represent well the data that we want to classify. This is why Holland's genetic algorithm was used to choose the location addresses that would be best to represent the input data classes.

---

* Doctoral degree study programme in field: Applied Informatics
Supervisor: Assoc. Professor Ladislav Hudec, Dr. Juraj Štefanovič, Institute of Applied Informatics, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 423–424.

In the beginning, the location addresses are filled with random bits. After training each of the input classes, we compute fitness function for all of the location addresses. Fitness function states, how good is the location at representing input data. Then the location addresses with the lowest fitness function are replaced with the crossover or mutation of the best addresses. Using this algorithm, locations after multiple generations should evolve towards the ones that are best to represent input reference addresses and corresponding input data. Other consequence is that after training, the reference address on the output of memory is also moving towards the address more representing input data. The output of counters effectively average the data given in input, so the reference address is in the "middle" of classified data.

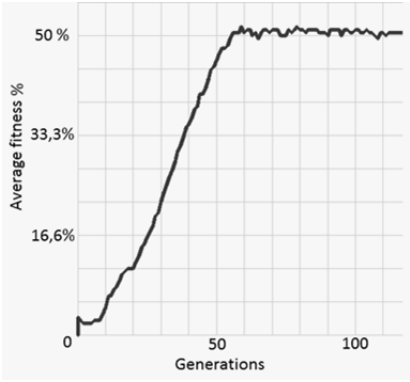


Figure 1. Graph of average fitness rising after generations of genetic enhancement

If we want to use SDM to create and train visual words, first we must roughly choose the reference addresses that can be than used to train SDM. That can be done using clustering algorithm like K-means. After they are trained to SDM and multiple generations of genetic refinement are used to create the best possible locations, to classify this particular codebook. If this is done, when we try to read from trained SDM, we need to present it with input bit vector obtained from the classified point in image. However this kind of memory after fixed number of steps, which is lower than the number of visual words, gives us only a bit vector, which is a reference address. But for the purpose of visual words, we need to get number of the reference address, not the complete vector.

To this end, we could use simple binary tree. Every level of the tree should belong to one of the location addresses. The leaves of the tree will be marked with numbers of reference addresses. That means, that to assign vector to one of the classes (visual words) we still need only fixed number of steps, that is number of location addresses. This can be a significant save of time, if the number of visual words is large, because number of location addresses << number of visual words.

In the preliminary experiments done with randomly generated binary vectors, using 100 location addresses and 1000 reference addresses, we were able to observe memory fitness rising to optimal level after just 70 generations of genetic enhancement.

References

[1] J. Sivic and A. Zisserman: Video Google: a text retrieval approach to object matching in videos. *Proc. Ninth IEEE Int. Conf. on Comp.Vision*, no. Iccv, vol. 2, 2003, pp. 1470–1477.

[2] a. Alahi, R. Ortiz, and P. Vandergheynst: FREAK: Fast Retina Keypoint. *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 510–517.

[3] D. Rogers: Kanerva's sparse distributed memory: *An associative memory algorithm well-suited to the Connection Machine.* 1988.

User Modelling Based on Tabbed Browsing: Browsing Scenarios as a New Source

Martin LABAJ*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
labaj@fiit.stuba.sk

Extended abstract

Current browsers support multiple means of browsing the webpages concurrently in multiple windows and tabs. A user can have multiple webpages opened at once and switch between them. It is established that this behaviour is common amongst users in various use cases, e.g. browsing the search results [2] and that the parallel browsing (tabbing) models better line up with the real user behaviour than the previous linear models [4]. It was proposed that such user behaviour can be leveraged in the adaptive web in multiple ways – from more accurately estimating webpage visitation and revisitation, to improving personalized recommendation by recognizing user tasks or sessions. In our work, we focus on building a user model based on tabbed browsing scenarios, modelling the user interests, goals and tasks.

The user modelling in our method is comprised of three steps: 1.) Acquisition and modelling of the basic tabbed browsing actions (e.g. the user has opened a link to a new tab and switched to it immediately), 2.) Modelling tabbed browsing scenarios from the tabbing actions (e.g. the user is keeping this tab opened as a reminder), 3.) Building/augmenting the user model (e.g. the user will be interested in concepts… with strength/probability of …).

For the first step, modelling the tabbing itself, we proposed a model consisting of user actions of opening webpages in various ways, closing them and switching between them, and an algorithm for model construction [3]. The actions are sourced from events recorded either in a web application via scripts included in a page (covering only single web application, but allowing to model every visitor), or in a browser via browser extension (covering only selected users, but allowing to model tabbing amongst heterogeneous web applications).

The second step, modelling the tabbing scenarios, is the core of the method. The users do perform tabbing in various situations [1]: doing *reminders*, *opening links in background*, *multitasking*, *going "back and forth"*, having *frequently used pages* ready, creating *short-term bookmarks*. These situations are however described from the users' point of view and observing those does not always have a feasible or unambiguous translation into user intentions. Therefore we define tabbing *scenarios* based on such situations in parallel browsing, which can express user interest:

---

* Doctoral degree study programme in field: Software Engineering
Supervisor: Professor Mária Bieliková, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

- *Retention of a tab* (retention behaviour) – The user keeps a tab opened for future reference, either for a *single future use* (e.g. a product page in an online store is retained until the user buys the item) or *recurrent future use* (e.g. a page with developing information such as weather forecast is kept open for quick access).
- *Opening links in background and exploring them* (examination behaviour) – The user opens multiple links from a single page or a group of pages and starts visiting them, closing some of them (not related/interested), retaining some of them and eventually possibly closing all of the spawned tabs or retaining one or more.
- *Changing context* – The user opens and switches tabs in order to work on a new task, e.g. the user is interrupted and wants to lookup information unrelated to his current activity.
- *Comparing content* – The user repeatedly switches within a group of tabs, e.g. checking multiple approaches to a programming problem from various sources (forum, documentation, etc.).

The tabs and pages opened in the tabs can belong to multiple scenarios gradually or at the same time, e.g. product pages retained for future use are now being compared by the user.

The user model is then built using these described scenarios as a base for interest estimation. The scenarios relate to current or future interest in given tabs (retention, opening links in background and exploring them, comparing content), aggregate the tabs into groups of interest (opening links in background and exploring them, comparing content), and switch which interest group is currently active (changing context). The interest in a tab means interest in a page opened in this tab and in concepts presented on such page. By tracking tab scenario memberships over observed time, grouping the tabs in a given time and activating the current interests groups, we build user model expressing user's *current* interest, possible *future* interest and *active* interest on top of automatically extracted concepts.

The proposed user model should bring improvement over the traditional techniques, such as tracking the time spent on a page or estimating the context from search queries, by more accurately and extensively modelling the user interests. The presented user model can find use in various adaptive features, such as personalized recommendation. In our future work, apart from improving the user model, we would like to focus on domain modelling as well – keeping pages opened and switching between them not only expresses user's interests, but also relations between the pages themselves, e.g. one page relates to another, explains it, or extends it.

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/0675/11.

References

[1] Dubroy, P., Balakrishnan, R.: A Study of Tabbed Browsing Among Mozilla Firefox Users. In: *Proceedings of the 28th international conference on Human factors in computing systems – CHI '10*, ACM Press, New York, New York, USA, (2010), pp. 673–682.

[2] Huang, J., White, R.W.: Parallel browsing behavior on the web. In: *Proc. of the 21st ACM Conf. on Hypertext and Hypermedia*, ACM Press, New York, USA, (2010), pp. 13–17.

[3] Labaj, M., Bieliková, M.: Modeling parallel web browsing behavior for web-based educational systems. In: *2012 IEEE 10th Int. Conf. on Emerging eLearning Technologies and Applications (ICETA)*, IEEE, (2012), pp. 229–234.

[4] Viermetz, M., Stolz, C., Gedov, V., Skubacz, M.: Relevance and Impact of Tabbed Browsing Behavior on Web Usage Mining. In: *2006 IEEE/WIC/ACM Int. Conf. on Web Intelligence (WI 2006 Main Conf. Proc.)(WI'06)*, IEEE, (2006), pp. 262–269.

Semantic Wiki for Research Groups

Martin MARKECH*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
matomarkech@gmail.com

Extended abstract

The content published on Web is constantly growing and Web is becoming difficult to process. It is because of many reasons. Data and resources are published on Web in raw formats (CSV, XML), without structure and semantics or also often in proprietary formats. HTML language describes more how the document should look like and hypertext links do not contain information about role in interconnected documents [1]. Aim of the Semantic Web is to solve these problems.

Linked Data initiative can be described as a set of the best practices for sharing and publishing information on the Semantic Web. This is in particular, relevant to researchers who can more effectively interconnect if they publish their content semantized. One of the paths to semantics utilization is the replacement of traditional wiki systems with semantic wiki systems.

The first semantic wiki system was created in 2004 [2]. Many new semantic wiki systems were created since then, but there are still some open issues, which we are facing [3]. Not all semantic wikis allow RDF import, so ontologies cannot be edited by user. Semantic wikis use URI of a page as dereferenced URI, which cannot be modified in the future. It is not problem on websites of encyclopedic type, but it is problem when we want to create deeply nested menu structure with more pages for one entity. Although many semantic wikis try to help user with content creation, neither seems to assist with semantic extraction from text. Some of the early semantic wikis allow to add semantics only when the text of page was created at first.

Our motivation is to improve existing wiki at our university taking into account specific needs of academic research groups, especially our group – PeWe. PeWe uses this wiki for its presentation and also to self-organize members. Although there are many semantic wikis, we do not want to completely replace our specific faculty wiki system and we prefer to create extension.

Our method consists of several parts. At the beginning we analyse the structure of wiki content. Then we propose templates, which help user with writing repeated blocks of text with similar structure. Templates are divided by topic and by granularity from simple to advanced templates. Filling the templates helps user to keep the same text structure and to auto generate semantics.

It ensures, that the created text has properly defined semantics, because values filled in fields are inserted in accordance with pre-defined ontologies. User can modify the triplet's values at any time.

---

* Bachelor degree study programme in field: Informatics
Supervisor: Jakub Šimko, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

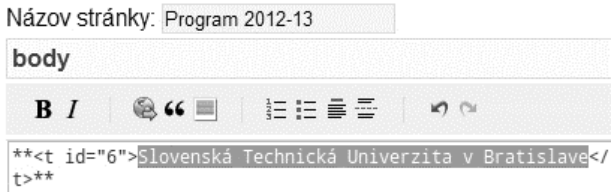


Figure 1. Markdown semantic marks.

Our method uses a triplet editor to give user the ability to change or add new semantics. Each semantic triplet has an attached ID. We use this ID in markup to connect text with semantic triplets. These triplets are stored outside of the markup – in semantic database Sesame. Since each wiki page has many revisions and allows creating multiple page parts, for the key value for context triplet field our method uses concatenation of page ID with revision ID and page part ID. Application for browsing semantics is independent and thus, we are not facing issues when the URI of wiki page is simultaneously dereferenced URI.

Second part of our method is generating semantic bibliography reference, because creation of correctly ISO 690 reference is not easy task. Digital libraries usually don't offer ISO 690 reference string. Mostly they offer BibTex format, which our method is parsing. The scenario is following:

1. User fill in some information about publication – DOI, authors, or title.
2. Webservice send query to Google in format *site:<digital library site name> <searched string>*.
3. Then it take the first result, expecting that it is the result with the best relevance.
4. Webservice loads the page in background and download BibTex data about publication.
5. Then it converts BibTex format to ISO 690 reference string.
6. User need to confirm the correctness.

Because these queries might be repeated, our method caches the results, which were confirmed as correct by user. If someone do query with equal values, the result will be fetched from cache.

Thanks to our extension, we have semantics of our wiki stored in a semantic store. We can connect to store endpoint and send SPARQL queries to get answers to complex queries. Semantics can then be re-used. Extension for semantics creation is browser independent and with a little extra work can be reused in different wiki systems.

We evaluate application with qualitative methods. Our hypothesis is that usability of application with new semantic extension is not too worse. We analyse the trade-off between the state without semantics – copypasting markdown and with new extension – templates with semantics.

Acknowledgement: This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/0675/11.

References

[1] Bizer, C. et al. Linked Data – The Story So Far. In Special Issue on Linked Data,

[2] Bry, F. et al. Semantic Wikis: Approaches, Applications, and Perspectives. In *Reasoning Web. Semantic*. 2012. pp. 329–369

[3] Maalej, W., Panagiotou, D., & Happel, H. (2008). Towards effective management of software knowledge exploiting the semantic wiki paradigm. Software Engineering, pp. 183–197. Retrieved from http://subs.emis.de/LNI/Proceedings/Proceedings121/gi-proc-121-022.pdf

SRelation – a Method for Relations Management and Navigation in Big Graph of Linked Data

Ján MOJŽIŠ*

Institute of Informatics
Slovak Academy of Sciences
Bratislava, Slovakia
jan.mojzis@savba.sk

Extended Abstract

Linked Data is a set of best practices for publishing and connecting structured data on the Web [1] based on Berners-Lee recommendations. Graph data can be represented using RDF model in one of formats (RDF/XML, N3, Turtle, etc.).
Graph traversal is problematic when using SPARQL [2], FILTER operation is heterogeneous in performance as triplestores vary. Likewise [3] compares triplestores against SPARQL queries, and differences are significant between stores. To help with navigation, SPARQL (specification) 1.1 added the option of property paths. Evaluations on graph traversal were performed [4] and the resulting performance is poor.

From the related work one can find Neo4j or SGDB (experimental store) [5]. In tests [6] SGDB was evaluated as best. SGDB uses key-value concept [5], Jena use b+trees with hash functions and clustering [7]. Unlike SGDB, our design (Figure 3) of store is based on b-trees indexation (instead of key-value pairs) of adjacency and the model is all-in-memory. Our solution is also prepared to be implemented as *distributed store*.

The motivation for us is Billion Triple Challenge (http://challenge.semanticweb.org/2012/) and the poor results of SPARQL 1.1 evaluations. In this paper we present SRelation as our store design and own graph traversal algorithm implementation. Our design supports 2 main operations needed for graph search and navigation, which is FILTER and property path search in SPARQL 1.1. These operations are not supported well by current triple stores such as Jena.

We used datasets ACM[1] and Wikipedia[2]. We split each dataset into smaller parts based on lines count (format N-triples) to get 10%, 30%, 50% and 70% sizes. We selected random 10,000 subjects from each type of dataset (ACM, Wiki) and 3 most frequent predicates (Wikipedia only 1). For each dataset and its size we then run SPARQL 1.1 property path traversal (Figure 2) on each of 10,000 subjects. The levels were 2 and 3. We used FILTER to filter-out duplicates. After we evaluated Jena, we focused on SRelation and executed compatible query with our breadth-first traversal for the same datasets, sizes and levels. We assured that the data returned matched. SRelation was able to perform traversal faster than Jena (Figure 1). We do not count time needed to translate SPARQL syntax (Jena).

---

* Doctoral degree study programme in field: Applied Informatics
Supervisor: Dr. Michal Laclavík, Institute of Informatics, SAS in Bratislava

[1] http://acm.rkbexplorer.com/models/dump.tgz

[2] http://downloads.dbpedia.org/3.8/sk/page_links_en_uris_sk.nt.bz2

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 429–430.

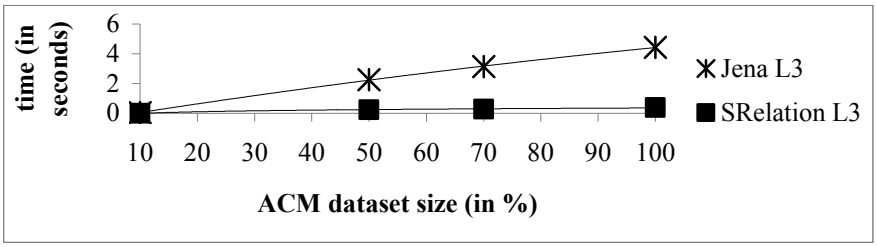


Figure 1. SRelation and Jena query-execution time comparison for ACM dataset, level 3.

```
SELECT ?property where {
<http://dbpedia.org/resource/Isaac_Newton>
<http://dbpedia.org/ontology/wikiPageWikiLink> /
<http://dbpedia.org/ontology/wikiPageWikiLink> ?property.
FILTER (<http://dbpedia.org/resource/Isaac_Newton> != ?property)}
```

Figure 2. Sample query for 1 (of 10,000) subject Isaac Newton, Wikipedia dataset and level 3 property path.

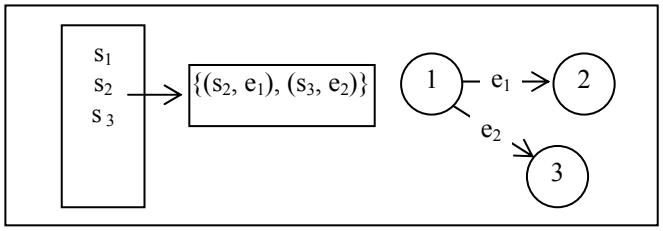


Figure 3. Architecture of SRelation's triplestore.

References

[1] Berners-Lee, T.: *Design Issues, Linked Data*. available on-line at: http://www.w3.org/DesignIssues/LinkedData.html, 13.3.2013.

[2] Schmidt, M., Hornung, T., Lausen, G. and Pinkel, Ch.: SP^2Bench: A SPARQL Performance Benchmark. *Proc. IEEE International Conference on Data Engineering (ICDE '09)*. IEEE Computer Society, 2009, pp. 222–233.

[3] Bizer, Ch., Schultz, A.: The Berlin SPARQL Benchmark. *International Journal on Semantic Web & Information Systems*, vol. 5, issue 2, 2009, pp. 1–24.

[4] Arenas, M., Conca, S. and Pérez, J.: Counting beyond a Yottabyte, or how SPARQL 1.1 property paths will prevent adoption of the standard. *Proc. World Wide Web (WWW '12)*. ACM, 2012, pp. 629–638.

[5] Ciglan, M. and Nørvåg, K.: SGDB – Simple Graph Database Optimized for Activation Spreading Computation. *GDM 2010*, DASFAA Workshops, 2010.

[6] Ciglan, M., Averbuch, A. and Hluchý, L.: Benchmarking traversal operations over graph databases. *Proc. GDM'12*, ICDE Workshop, 2012.

[7] *Jena architecture*. available on-line at: http://jena.apache.org/documentation/tdb/architecture.html, 14.3.2013.

[8] Martínez-Bazan, N., Gómez-Villamor, S. and Escale-Claveras, F, "DEX: A high-performance graph database management system," GDM 2011, ICDE Workshops, 2011.

Metadata Collection for Personal Multimedia Repositories Using Games with a Purpose

Balázs NAGY*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
chelseadrukker@gmail.com

Extended abstract

With the increasing number of personal albums and photos in them, users have more and more problems with their organization [4]. This is due to lack of descriptive data, however, their amount only depends on the abilities and will of the image owners. Tools for metadata creation are available – the main problem is with the user motivation: because tagging and annotating of photos is usually a boring activity and its execution takes extended time periods [5]. Other methods for obtaining metadata to general images also exist (automated methods, crowd-based, games with a purpose [1]), but these are unable to deliver specific metadata needed for personal imagery (e.g. names of persons).

Our aim is to create a method to enrich personal photo albums with keywords and also named entities. To achieve this, we use tool that imports personal albums, allows creating annotations using a GWAP, extracts keywords and named entities from annotations and allows browsing in albums using obtained metadata.

Earlier, we devised a game with a purpose called PexAce [2, 3] for harvesting textual annotations to general images. Earlier experiments showed that people playing with their own photos are more engaged to game and also interested in creating annotations. Another side effect is that these annotations are more precise and relevant for the owners of these photos. By merging our game with automatic approach of metadata acquisition from game-produced annotations, we found an appropriate solution for creating metadata for personal photo albums. The main contribution of this work is a framework for processing annotations written to personal photos. This framework is based on metadata extraction modules called extractors and is extensible from this perspective.

Inputs of all extractors are following five entries: photo, album to which photo belongs, user who created the annotation, annotation, and timestamp. In addition to these information, extractors have access to all logs saved during games and use them to refine results of annotation processing. Extractors are divided to two groups (Table 1) depending on their output. While the first group includes extractors extracting keywords without specifying their types or any other information about them, the extractors in second group are extract typed keywords and also named entities such as persons, geographic locations, events or holidays.

Master degree study programme in field: Software Engineering
Supervisor: Jakub Šimko, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 431–432.

Table 1. Grey: extractors without specifying type (all combinations of preprocessing methods with candidate selections). White: extractors specifying type (all combinations of entity lists with comparing methods and 3 existing tools).

Preprocessing (Naive, Alchemy API, Tagthe.net, Zemanta API)	X	**Candidates selection**
Lists of entities (Friends, events, places, holidays, etc.)	X	**Comparing methods** (Levenshtein, Hamming, Jaro-Winkler distance)
Existing extern APIs (Alchemy API, Tagthe.net, Zemanta API)		

To aggregate outputs of extractors we designed two types of aggregators for each type of extractor. These contain information about credibility of extractors and use it for aggregation of the results. Credibility of an extractor depends on the particular method (pre-processing, candidate selection or comparison method) used by the extraction (e.g. results provided by a particular tag extraction API can rank higher than results of another one).

To evaluate our method, we implemented different parts of our solution in a particular order. In the first stage we re-implemented and re-designed the PexAce game, which is now more user friendly and as a web application it can be run on multiple platforms. Then we implemented importing tools to transfer photos into our database. After this we had the first opportunity to evaluate its functionality and realized a qualitative verification in the form of an interview with a small number of users. In second phase we implement extractors with aggregators (processing annotations written to photos) and photo gallery (exploiting keywords extracted with our methods). The photo gallery offers the opportunity to verify usability of extracted keywords based on user feedback.

For quantitative verification of our method we created a tool which enables users manually annotate personal photos. Comparing these annotations with our results we can determine the precision and recall of the method we designed.

Analyzing the current state of solutions which are oriented to obtain descriptive data to photos, we imply the absence of solutions for personal photo albums. Because of this, and the lack of descriptive metadata in these albums we decided to use our existing method for metadata acquisition to general photos, and redesign it for obtaining metadata to personal photos.

Acknowledgement: This work was partially supported by the Slovak Research and Development Agency under the contract No. APVV-0208-10.

References

[1] von Ahn, L., Dabbish, L.: Designing games with a purpose. *Communications of the ACM* 2008. Vol. 51, no. 8, pp. 57.

[2] Nagy, B.: Acquisition of Semantic Metadata via Interactive Games, *Proceedings of IIT.SRC* 2011. Vol. 1, pp. 9–15.

[3] Simko, J., Bielikova, M.: Games with a Purpose: User Generated Valid Metadata for Personal Archives. *SMAP 2011*, Sixth International Workshop, 2011. pp. 45–50.

[4] Vainio, T. et al.: User needs for metadata management in mobile multimedia content services. *6th Int. Conf. on Mobile Technology, Application & Systems*. ACM, 2009. pp. 51.

[5] Wenyin, L. et al.: *MiAIbum – A System for Home Photo Management Using the Semi-Automatic Image Annotation Approach*. 2000. pp. 479–480.

Software Transactional Memory for Peer-to-Peer Systems

Aurel PAULOVIČ*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
paulovic@fiit.stuba.sk

Extended Abstract

Transactional memory [4] is an abstraction that tries to use the notion of a transaction as a programming language construct for concurrency control. The idea of transactions is at its core very simple. A program can wrap a block of computation into a transaction, making it execute in respect to other simultaneously running processes as a single indivisible *atomic* operation that is completely *isolated* from other concurrent transactions. In other words, although a process executes its instructions in a transaction as a sequence of steps that create and mutate possibly shared variables, other processes should be allowed to see the state of the running program only as it was either immediately before the start of the transaction or immediately after successful execution of the transaction, thus not the intermediate steps.

The goal of transactions is to provide a simple and convenient way to deal with concurrent access to shared data while being immune to data races and deadlock and allowing composability. However, the precise semantics of atomicity, isolation, data versioning, conflicts and failure atomicity are convoluted and can differ in respective implementations of TM. In addition, although similar to and inspired by the transactions in database systems, transactional memory differs in several key aspects that prohibit straightforward adoption of database transactions to TM, the most notable being the difference in workloads, relative overhead cost and the existence of non-transactional data access.

Traditionally TM implementations detect conflicts between transactions based on the reads and writes of individual memory addresses or the individual variables and objects described by these addresses. Two transactions are said to be in conflict, if one transaction tries to write to the same address that the other concurrent transaction has read or written to. While such approach allows us to design a general transactional memory runtime without any knowledge of the operations performed in transactions and their underlining data structures, it also decreases the amount of concurrency that the system can support. In order to alleviate this issue the TM can use *higher-level conflict detection and resolution* (HLDCR) that detects conflicts not using the accessed memory addresses but the semantics of the operations performed on data structures instead. For the transactional memory to be able to support this, the TM runtime has to use *abstract data types* (ADT). ADTs are required

* Doctoral study programme in field: Software Engineering
Supervisor: Dr. Peter Lacko, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 433–434.

to completely encapsulate the shared state that they manage and declare the mutual commutativity and its constraints as well as the inverses of the operations that they provide. HLCDR in TM [2, 3] has been used so far only in non-distributed systems and, to our knowledge, it has not yet been studied in the context of improving fault tolerance of transactional memory.

Despite the years of research of transactional memory systems many problems still remain open and prohibit widespread adoption of TM in practice. We focus our efforts on tackling one of the issues, namely the design of an efficient fault tolerant distributed software transactional memory suitable for peer-to-peer systems.

Since there is no way to prevent node failures in distributed systems, the only way for transactional memory to remain consistent and available is to replicate its shared data. The TM has then to synchronize the data replicas, which has negative effects on the performance and amount of communication in the system. We want to lower the need for data synchronization using the abstract data types and higher-level conflict detection and resolution, which are in part interchangeable with conflict-free replicated data types, which are commutative and convergent. We believe, that ADTs and HLCDR could allow us to employ a form of lazy replication using that we could defer and piggyback the propagation of data updates on other inter-node communication. This would speed up the validation and commit phases of the transactions. In addition, using commutative non-conflicting operations could help us achieve fault tolerance in that the we would be able to reconcile and merge different versions of data after a node failure.

Strong consistency and isolation semantics of transactions combined with the implication of CAP theorem in distributed TM systems often limit the practical implementation of TM. Existing transactional memory designs favor consistency over availability and partition tolerance; however, we would argue that the characteristics of P2P systems require a different trade-off. Following the use of eventual consistency (EC) in recent NoSQL data stores we believe that an eventually consistent transactional data model similar to the EC transactions for concurrent revision proposed by Burckhardt et al. [1] could be adapted to distributed transactional memory systems. EC would require the TM to host a mechanism that would be able to resolve conflicts between data replicas after some time period, which can not be handled by the usual transaction rollback semantics. It could, however, be conveniently solved by the use of abstract data types that could in addition to minimizing the amount of conflicts also define merging rules for their operations.

We believe that the design of a fault tolerant distributed software transactional memory for unreliable commodity networks with higher latency requires us to focus on minimizing the needed amount of replica synchronization as well as the conflict-rate of transactions. We propose to achieve this goal by the use of a combination of higher-level conflict detection and resolution using abstract data types and the overall relaxation of consistency model using eventually consistent transactions.

Acknowledgement: This work was partially supported by the Slovak Research and Development Agency under the contract No. APVV-0233-10.

References

[1] Burckhardt, S., Fahndrich, M., Leijen, D., Sagiv, M.: Eventually Consistent Transactions. In: *Proceedings of the 22n European Symposium on Programming (ESOP)*, Springer, 2012.

[2] Herlihy, M., Koskinen, E.: Transactional boosting: a methodology for highly-concurrent transactional objects. In: *Proceedings of the 13th ACM SIGPLAN Symposium on Principles and practice of parallel programming*. PPoPP '08, New York, NY, USA, ACM, 2008, pp. 207–216.

[3] Koskinen, E., Parkinson, M., Herlihy, M.: Coarse-grained transactions. In: *Proceedings of the 37th annual ACM SIGPLAN-SIGACT symposium on Principles of programming languages*. POPL '10, New York, NY, USA, ACM, 2010, pp. 19–30.

[4] Larus, J., Kozyrakis, C.: Transactional memory. *Communications of the ACM*, 2008, vol. 51, no. 7, pp. 80–88.

Innovative Platform-Independent VPN Client

Vladimír RUMAN*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
xrumanv@fiit.stuba.sk

Extended Abstract

The requirement for reliable and secure access to internal corporate networks is currently commonplace. Remote access to the private network is widely used in enterprises, offices, academic field. VPN technology provides a connection between the client and remote internal network that is secure, reliable and allows wide range of utilization. There are several approaches which provide a given functionality. IPsec and SSL VPN are in general the most commonly used solutions.

One of the complex open source solutions is Adito VPN. It is a platform-independent web based SSL VPN solution which is based on client server architecture [1]. It is a so-called clientless solution so there is no need to install any client. It offers several levels of access to the internal network. It is therefore preferred solution anywhere it is necessary to have a flexible solution that is available from any place. A disadvantage of the Adito VPN is no support for full access to the internal network. Adito contains a module called Agent that is downloaded to the client device in the form of an applet. Agent allows port forwarding and other services. Both server and Agent are designed to allow the simple adding of new features without the need to interfere with the existing structure. Module Applications can extend the Adito with the new features in the form of completed and independent applications.

In this paper we proposed the extension of Adito VPN. This extension allows full access to the internal network in conjunction with a web based VPN which can work without need to install a client on user devices. However, our solution can be used as a native client, too. This allows easy use in mobile platforms. Extension consists of the three parts:

- UDP tunnel which is connection between client and server.
- Communication protocol which handle control connection between both sides.
- Authentication scheme which issues certificates and keys.

Tunnel is essential for full access to the internal network. The tunnel is secure environment where data can flow through. Unlike IPsec which modifies the IP stack in operating system kernel, SSL VPN uses virtual network adapter to create tunnel. The adapter seems to be real to the operating system but unlike the physical interface, which sends data to network, the virtual adapter sends data back to user space. To create such interface is used TUN / TAP driver [2].

---

* Master degree study programme in field: Software Engineering
Supervisor: Peter Vilhan, Institute of Applied Informatics, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 435–436.

In the proposed extension, connection between the client and the server represents UDP tunnel. It is established in the initial connection phase (handshake). Not only data passes through but also control connection (which is secured by TLS protocol) is created within tunnel (Figure 1). Handshake is temporarily protected by asymmetric ciphers. During the initialization phase when connection is being established no data can be send. UDP tunnel in initialization phase is used only for embedded TLS connection to ensure authentication, integrity and confidentiality of the connection. A type of encryption mechanism is the result of negotiation between both sides. Negotiation takes place within the control connection after a successful TLS channel establishment and mutual authentication. Routing information about internal network is sent to the client via the control connection. Secure UDP tunnel is established after successful negotiation of encryption mechanism. Data can flow through tunnel until connection is broken.

We created our own communication protocol which is used for exchange information between client and server. Protocol is very simple. Every message has a byte header and body. Header specifies type of message. The protocol allows sending messages in form of key-value pair. Due to this fact there is no need to precisely specified content and length of information in the protocol specification. It brings a significant flexibility which is necessary for developing the clients for mobile platforms.

Authentication is also very important. Adito communicates with user through HTTPS protocol. Certificate is needed only on the server side. Users usually use their login and password for authentication. HTTPS connection will be used for our solution. Through this connection we can transfer the client certificate with private key. This certificate is used to authenticate the client to the server when tunnel is being created. Certificate is valid until connection is broken and Agent is removed from client side. A certificate is issued for each instance of the client so the new certificate must be issued when connection is reestablished. However, our solution is not limited to Adito VPN thus it must be capable to authenticate VPN clients for mobile platforms, too. Because of this the server must be able to issue certificate which is bind to particular user.

Proposed solution extends Adito VPN with useful and very important feature. In the future work, we plan to perform stress tests to demonstrate secureness and effectiveness of the proposed solution.

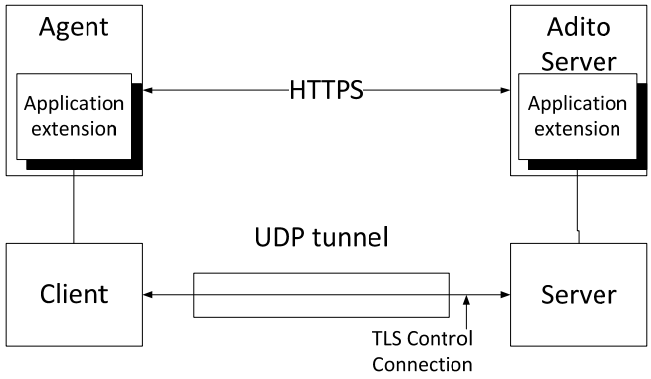


Figure 1. Component view of the Adito VPN extension.

References

[1] AditoVPN, What is AditoVPN.[Online; accessed February 16 2013]. Available at: http://sourceforge.net/apps/trac/openvpn-als/wiki/what_is_openvpn-als

[2] KRASNYANSKY, M.:TUN/TAP Universal Driver.[Online; accessed February 16 2013]. Available at: http://vtun.sourceforge.net/tun/faq.html

PNets – the Verification Tool Based on Petri Nets

Miroslav SIEBERT*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
siebert@fiit.stuba.sk

Extended abstract

Sequential logic circuits design is based on finite state automata that can be represented by Petri nets. Principles of Petri nets belong to basic knowledge of designers and have to be involved in an educational process. Besides of theoretical knowledge in the educational process there are suitable practice examples to demonstrate Petri nets design and to show their behaviour. Therefore the goal has been to develop interactive software tool for understanding and using Petri nets in the digital design process. This tool should offer various possibilities, especially when marking of places are changed and an error can occur. Verification of the net properties and characteristics can be realized by this educational tool. The new proposed and implemented educational tool for Petri nest is enriched by useful features such as scheme save and later edit, save them as picture, send them by e-mail or they can be involved into documentation.

A new method was proposed and implemented into an educational tool called PNets (Petri Nets). In comparison with existed similar educational tools [1], the proposed and implemented PNets is simpler, intuitive, less demanding on hardware and portable with detailed verification of the designed Petri net. This tool offers modelling and functionality animation of designed Petri nets together with verification of their fundamental properties as safety, liveness, conservativeness, boundedness, contact-free and reversibility.

PNets operates in two modes. The first one is the editing mode which is used for creating of a net and its subsequent editing. The second one is aimed to simulation. This mode starts by switching the mode selector button from the state “off” to state “on”. Transition is executable only in the simulation mode. Executable transition is depicted in blue. It is possible to analyze the characteristics of the current generated net in both modes. Analysis of the properties is provided by using the reachability tree.

The PNets educational tool has been developed using object-oriented approach of Java programming language which ensures its portability and operating system independence. The number of settings and options from a user are minimized in order to use this tool as simple as possible. Several tools parameters have preselected default values (e.g. screen resolution) and are adjusted automatically. The educational tool is active only in case of an active interaction from the user otherwise it is in the "sleep" state. Therefore this tool has a minimum hardware requirements

---

* Doctoral degree study programme in field: Applied Informatics
Supervisor: Dr. Jana Flochová, Institute of Computer Systems and Networks, Faculty of Informatics and Information Technologies STU in Bratislava

for its execution and it is therefore well suited to run also on older computers, mobile devices and tablets. [2]

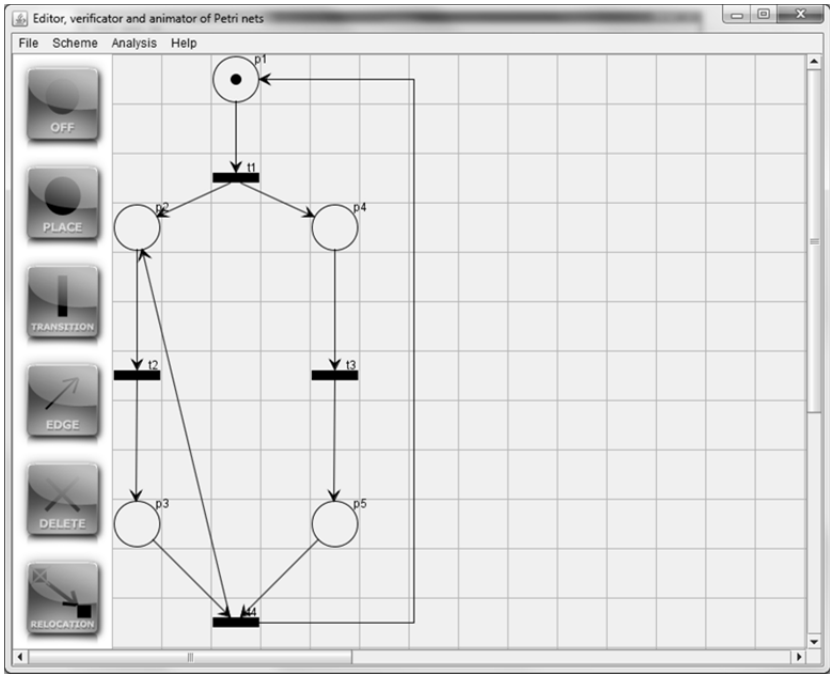


Figure 1. Tool graphical user interface with example.

Figure 1 illustrates the layout of the tool's graphical user interface. Screen elements layout is chosen on the basis of already existing tools listed in [1]. Control buttons will perform the main service runtime. They are placed on the left of the screen. The reason for the vertical placement of the buttons is also a growing number of wide displays on personal computers, mobile devices and tablets.

Properties of the nets were tested by more than hundred different examples e.g. by example in Figure 1. This net is live because it's always possible to reach a marking which enable to trigger all transitions. It is not bounded because the number of token in the place *p2* is increasing to infinity. Therefore the net is neither consistent, nor conservative, nor safe. There is no place with setting the maximum number of tokens therefore the net is contact-free.

This paper describes the design and implementation of the new educational tool called PNets. The main benefits of PNets are its portability, simplicity, current (running on the latest operating systems) and detailed verification of Petri nets properties.

Acknowledgment: This work was partially supported by the Slovak Science Grant Agency (VEGA 1/1008/12 "Optimization of low-power design of digital and mixed integrated systems").

References

[1] Heitmann, L., Moldt, D.: *Petri Nets Research Groups*, University of Hamburg. [Online] 2013. http://www.informatik.uni-hamburg.de/TGI/PetriNets/research

[2] Siebert, M.: *Editor, animátor a verifikátor Petriho sietí,* Master's thesis, STU FIIT, Bratislava, (2011), 80 p.

Personalized Recommendation with Considering of Social Aspects

Jakub ŠALMÍK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
som@jakubsalmik.com

Extended Abstract

Social media make large expansion last time and we live today in time where dominating relationships created on social networks and where we share informations with ours on-line friends.

We propose a method for content-based recommendation which work with informations about users obtained from social media and using them for recommendation in domain of on-line newspapers. One of the most important factor obtained from social networks are trust of user, which can help us filter fake informations.

Our method work in three base steps: creating a user model from social network activity, search of right article and recommendation based on previous steps (see Figure 1).

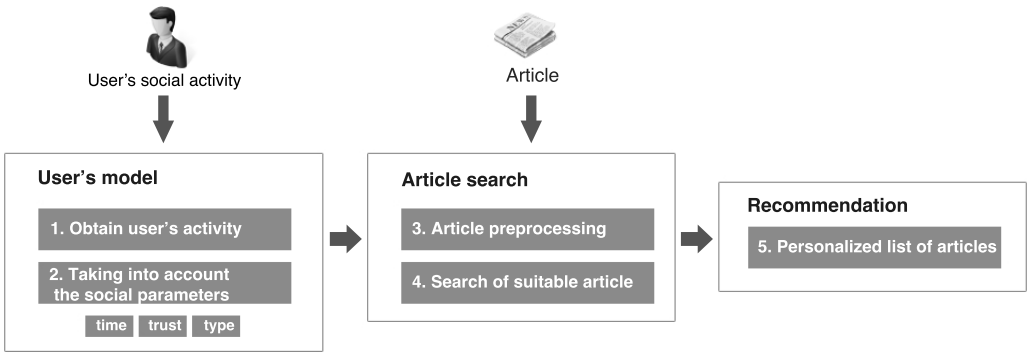


Figure 1. Our method for getting data and recommendation.

There are some approaches to find user interest on social media. Two ways are interesting for us: user tagging items on social networks and user activities on social networks. First approach depends on user's interest of tagging items while second using natural user's activities on social media.

* Master study programme in field: Software Engineering
Supervisor: Michal Kompan, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

Accompanying Events

TP Cup – The Best Student Team Competition Showcase at IIT.SRC 2013

Mária BIELIKOVÁ*

Slovak University of Technology
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
bielik@fiit.stuba.sk

Abstract. Best student team competition TP-CUP is organized fifth time this year. The competition is aimed at excellence in development information technologies solution within two semester long team project module in master degree programmes. This year 11 student teams presented in IIT.SRC 2013 showcase their projects. Key concepts of their projects are included in following sections of the proceedings.

1 Background of the Competition

Team projects play an important role in the education of engineers. Team projects have a long tradition in informatics and information technologies study programmes at our university. Module firstly named *Team project* was introduced in the academic year 1997/1998 in software engineering and in subsequent years it was adopted as compulsory module for all master degree students. Its intake is each year 25-30 teams of 5-7 students in all study programmes. The main objective is to give students a hands-on experience with different aspects of working in team on a relative large task.

In designing a team project as a part of a curriculum, we considered several aspects or different alternatives to particular issues such as team formation, team communication methods, team assessment, problem assignment, development process and team supervision. Our experience with such projects is that a satisfiable solution (in terms of the team project objectives, i.e. experience with different aspects of working in team on a large problem) requires time longer than one term, so we designed our team project as two semester module. Supervisors who are available (either academic staff or an industry partner) determine problems being solved. Teams consist of 5-7 students. They are created under our active control. Our criteria aim at balancing differing specific knowledge of team members and different experience in various team roles. We also respect the students' preferences to some extent (a student can specify one student to become a member of the same team).

We let teams bid for problems proposed by supervisors. A competition between teams is established and students have opportunity to exercise writing and presenting the bid. The students bid with their knowledge, skills and achievements related to the selected problem, and with

* Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

a preliminary sketch of solution based on the open question-answer session with a customer (mostly a supervisor).

Although the quality of the final result is an important measure of a success of a team, we markedly concentrate on the process applied. We adopted the development process with at least two iterations. Several teams use agile development methods each year (e.g. this year we have teams working according the Scrum methodology).

The amount of freedom and supervision should be balanced in order to create a true learning experience for students. To simulate the reality, students should have a considerable amount of freedom. On the other hand, since students usually have no or just little project experience, some amount of supervision, monitoring and guidance is needed to ensure sufficient progress and a successful result. In order to reach balance between freedom of students and supervision we specify in advance certain requirements on the content of documentation to be produced. Students have to prepare and follow a detailed project plan. We prescribe certain parts of the project plan, such as list of activities, milestones, dependencies, and responsibilities according to established team process. Students are free to define the activities that are necessary to successful accomplishing of the project. We accompany the Team project by lectures on project management, teamwork, and quality assurance.

2 Stages of the Best Student Team Competition

In order to emphasize excellence of the students' teams we established the Best Team Competition called TP Cup in academic year 2008/2009. The competition is aimed at excellence in development information technologies solution within our two semester long team project module in master degree programmes.

The competition has several stages. It starts with an application in the middle of the first semester. First stage finishes by the end of first semester when the teams submit interim report. We filter out teams which do not fulfil basic criteria on quality of work performed. Second stage culminates in the middle of second semester when students submit key concepts in form of two page report into IIT.SRC proceedings and present their projects in the TP Cup showcase organized as a part of the IIT.SRC conference. This year 11 students' teams presented in form of showcase their projects at the IIT.SRC 2013. Third stage presents finalizing the projects. It ends by our grand finals where board of judges consisting experts from industry selects the winner team which lands the challenge cup – "*Best FIIT Student Team of the Year*".

More information about TP Cup can be found on the Web:
`http://www.fiit.stuba.sk/tp-cup/`

Discovering and Evaluating Relations in the Field of Science and Research

Michal ADDA, Dávid BADO, Miroslav BLŠTÁK, Marek TOMČO,
Anton SZORÁD, Martin UHRIN , Tomáš ZBOJA*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
tim03.fiit@gmail.com

Databases of research publication that are in a Slovak environment created mainly by academic libraries and research institutes (e.g. Slovak Academy of Sciences) are a great information resource for evaluating of the current research state. The challenge for databases of research publication is searching for references, assigning them to the corresponding publication and evaluating them. Majority of systems is based on manual references processing – i.e. searching for references in a citation database interface and inserting them into corresponding research publication. This kind of processing is time-consuming and based on implicit experience and knowledge of human processors.

The aim of our project is to create a tool for automating processing related especially to processing of references. It solves problems with searching for references, where the correct identification of publication, the references of which are searched, is important. The next problem, which our project is trying to solve, is processing of references from heterogeneous resources (different citation databases). These resources offer references in a different format and bring duplicates of references. The important part of the project is to develop, to verify and to apply tools for data deduplication and normalization.

The expected outcome of our project is a complex system for publication and references management presented as a webpage (DIGLIB). Therefore, the project is divided into several partial tasks.

The first partial task is to create a system for a user registration. The user has to be registered in order that we are able to obtain basic information needed for finding references.

The second partial task is to create a database of research publications; it means the database of records, to which references will be added. This database contains bibliographic data and is created by importing data from existing databases of the research publication. The source format is XML CREPČ, which is a standard in Slovakia, and guarantees compatibility with all databases of research publication of Slovak universities. The import of data is realized by using files in this format. In the future the project can be extended on implementation of protocol OAI PMH, which enables automatic harvesting of new publication records. Alternatively, there can be used protocol Z39.50, whereby our system can connect to a publication evidence system and new records can be found and stored according to data of the registered user. The implementation of OAI PMH and

* Master degree study programme in field: Software Engineering
Supervisor: Dr. Nadežda Andrejčíková, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 445–446.

Z39.50 can be included in the project only experimentally, because it is dependent on the cooperation with source databases.

The third partial task is data deduplication and normalization. The normalization is needed because input data come from different resources, which stores records in different formats. The project has to normalize data into a format, which corresponds with a data model of the system. The process of deduplication is based on evaluating similarity of records using algorithm, which uses a similarity method *n-gram similarity* [1] for evaluating.

The fourth partial task consists of pairing publications with reference resources (citation databases) and searching for references. We search publications by title and author's name of the publication (all variant forms of the name, which system had learnt until the moment, are taken in consideration). We use the method of the similarity evaluating for the pairing of publications. Based on this method, system determines if publications are paired. If a publication is paired, the system adds an identifier from the citation index to it. For these publications the system finds references. References are collected using identifiers gained by pairing. Subsequently, the next phase of data normalization, deduplication and also tagging autoreferences must be implemented.

The last partial task is to notify a user about new references which have been found. The system will notify the user by email, but the user will also be able to see new reference using a web interface of our project. If found references are faulty, the user will be able to identify them and delete them. By doing this, our system is learning and it will create new rules for finding new references. The knowledge base is being built, and it is able to process the knowledge of users and use it to increase the accuracy of results.

The correctness of the methods and tools used in our project for obtaining references for publications will be tested by comparing the results with those of manual searching and processing. The test data sample will consist of publications published over some period of time. It will contain publications without references, publications which have references with faults or publication the references of which have attributes missing. The system will find references for publications from data samples and these will be compared with manually found references. We assume that the system will be able to find at least the same amount of references as were found manually. It is probable that it will find even more, because our system anticipates the errors in data, it also uses alternative names of persons and organizations.

Tracking references for publications is important for assessing the results of science and research projects by using scientometric, bibliometric and infometric methods. We created the DIGLIB project to help the researchers. It will enable them to manage their publications and relieve them from gathering references to their work manually. Our system will allow them to spend more time on research and less time on administration. Our generated database of publications and references, together with our knowledge base, will be a good base for the application of assessment tools for research and development.

References

[1] Kondrak G.: *N-Gram Similarity and Distance. String Processing and Information Retrieval*, Springer, Berlin, 2005, pp. 115–126.

Re-Imagining User Interface

Ján ANTALA[×], Martin ČERTEK[×], Jakub GONDÁR[*], Ondrej GRMAN[×],
Silvia HUDAČINOVÁ[×], Michal IGAZ[*], Richard SÁMELA[×]

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
tim2_fiit@googlegroups.com

Computer vision has undergone remarkable progress in the last decade. And it steadily grows to even greater importance over time. This trend is result of this modern age. Kinect is widely used in many different areas and its usage in the game is loosely substituted by other forms [1]. Our aim is to spread knowledge of the Kinect and other technologies due to its rather narrow usage outside of gaming or research world. For majority of common people or gamers it is still easier to use more conventional inventions like keyboard and mouse for games and remote controllers for typical devices at home or workplace.

One of the interesting directions of use is in the intelligent houses. It is a direction of controlling various home appliances with natural user interface. [2] One of the key tasks of the project was the extension of the previous team project [4]. One type of input is not very reliable. Malfunction of the device could bring several problems. Another problem is when it comes to the disabled people. Deaf or blind people are not able to use conventional applications properly. Therefore we have conceived several ideas to achieve greater reliability and to provide easy-to-use application even for disabled people.

We continue with gesture recognition [4], but we focus on other input types and combinations of them. Namely mobile phone with operating system android, speech recognition also on the platform of android, infrared device and for enhancements in motion recognition provided by Kinect. This expansion of functionality allows us to achieve our most important goals. These are cross-platform application with multiple input types and control of common devices.

Application is created mainly for the experimental usage but all improvements were made with taking the common user into account. Core idea and reason for the development is to provide fully configurable application. User in this context is a researcher. He or she provides output, which is used by common user or end user in this meaning. The importance lies especially in the innovative improvements of another input device and in massive extension of output capabilities. This is very important especially for research what is our application primary made for. And this should contribute to faster and better spreading of the incorporated technologies together.

Structure of the system is composed of several modules. The meaning and function of each module are described in the next section:

---

[*] Master degree study programme in field: Software Engineering[*], Information Systems[×]
Supervisor: Dr. Vanda Benešová, Institute of Applied Informatics, Faculty of Informatics and Information Technologies STU in Bratislava

- *Input modules:*
  - *Kinect device* – this module provides functionality for correct handling kinect device and its input. It updates known gestures on the server, too. This module provides learning of the new gestures and editing of the existing ones.
  - *Mobile phone* – this module provides functionality for acquiring both touch gestures [3]and voice commands. It also keeps the server updated. This module provides learning of the new touch gestures and editing of the existing ones.
- *Central application:*
  - *Server* –this module is a backbone of our application. Provides functionality for keeping updated list of input and output commands and their matches according to the user's experiment. Configuration tools are available from all input devices and web browser.
- *Output modules:*
  - *Infrared device* – this module provides functionality for handling of the infrared device. It can be used for learning of the new set of transmission codes (TV and so) and for transmitting output commands. This is supposed to bring variability to controlling other devices. Infrared rays are able to control wide range of devices. It allows controlling even old devices (Television and so on).
  - *MAC computer* – this module provides functionality for controlling of pc. This module replaces old controlling with keyboard and mouse with other possibilities of controlling such as motion, speech or touch gestures of mobile phone.

We used several programming languages for our implementation. Server is running on the node js technology. Other modules use C, C++, Java, Python and other languages. We used Qt framework, Java and web technologies for GUI creation.

Our application is fully configurable and the most importantly modifiable. Whole process of the life cycle of our application was focused on this aspect. With the many loosely coupled modules and communication standard we were able to achieve significant progress of our goals. New modules are easy to add due to server concept and independent communication standard. That allows all future extensions or improvements maintain aboriginal architecture of application and prevent degradation of the structure.

References

[1] Francese, R.; Passero, I.; Tortora, G.: Wiimote and Kinect: gestural user interfaces add a natural third dimension to HCI. In *Proceedings of the International Working Conference on Advanced Visual Interfaces (AVI '12)*, Genny Tortora, Stefano Levialdi, and Maurizio Tucci (Eds.). ACM, New York, NY, USA, 2012, pp. 116–123.

[2] Jain, J.; Lund, A.; Wixon, D.: The future of natural user interfaces. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems (CHI EA '11)*. ACM, New York, NY, USA, 2011, pp. 211–214.

[3] Loureiro, B.; Rodrigues, R.: Multi-touch as a Natural User Interface for elders: A survey. Information Systems and Technologies (CISTI), *2011 6th Iberian Conference on* , vol., no., 15-18 June 2011, pp. 1–6.

[4] http://labss2.fiit.stuba.sk/TeamProject/2011/team11is-si/team11-11.ucebne.fiit.stuba.sk/artQ/index.html

OwNet Android

Jozef ARPÁŠ[*], Jaroslav RAIS[**], Marek LÓDERER[**]
Michal ROŠKO[**], Pavel RUŽIČKA[**], Vladimír SUDOR[**]

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
fiit.tim.06@gmail.com

1 Introduction

Nowadays, the Internet is the largest and most accessible source of information and is also becoming an important source of learning materials. Nevertheless, the Internet is not accessible to everyone even in the modern world. Many developing countries in Africa suffer from a intermittent, low-quality Internet connection, which reduces the possibility of using it efficiently for educational purposes in schools.

On the other hand, many African people have smartphones or tablets with Android operating system, which allows them to use a variety of Internet-based services. However, these people need a service which would help them to overcome issues with quality and stability of the Internet connection.

This is exactly what we are aiming at with our application *OwNet* for Android. Our vision is to develop a solution that turns any Android device into an OwNet node, providing its owner the experience of offline and collaborative browsing. We built *OwNet* for Android following the concepts and principles of a project *OwNet* [1] built at our faculty in previous year. While the previous version worked only on "classical desktops" connected to the LAN and the original design was tied to the place where the computer is located, the version which we propose comes with greater flexibility and follows the worldwide trend of mobile computing.

2 Application OwNet Android

We developed a complex solution which includes integrated SQLite database, SD card data storage and access, multithreaded processing and service-oriented interfaces. We provide a constantly running proxy service which can be setup as an ordinary proxy on one's favorite mobile browsing application (such as Opera Mobile). This design decision makes our application functional on all devices with Android 2.2. and higher.

The core processing is to catch a browsed web page and store it on an SD card for further retrievals along with appropriate metadata in the database. This is the way how a web page can be visited even when the Internet connection is currently unavailable.

---

[*] Master degree study programmer in field: Software Engineering
[**] Master degree study programmer in field: Information Systems
Supervisor: Dr. Michal Barla, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

2.1 Offline browsing

The application cooperates with all mobile web browsers which have an option to set up a proxy. The application captures the requests for a resource with a given URL address from the browser and executes them instead of the browser. Downloaded resource is sent to the browser while at the same time it is also stored on an SD card of the mobile device. The next time the same URL address is to be accessed, the resource is retrieved from the memory of a mobile device, not from the Internet. Benefit of the solution is that if the resource is already saved on the SD card the internet connection is not necessary. This method saves traffic and size of transferred data.

2.2 Ownet web portal and toolbar

Our OwNet for Android comes with a web portal and a toolbar included into each displayed page. These two tools allow for easy management of all the content within OwNet and bring collaborative aspects into ordinary browsing. User can not only browse through cached web pages but can also rate, tag and recommend web pages so that other users can find them more easily.

The portal also allows any user to upload a content into OwNet and thus make it accessible to anyone within a local network even if connection to the Internet is currently broken. This feature is especially useful for teachers, which can easily share study materials with students while having a certitude that the content will be accessible during lesson.

2.3 Distributed caching mechanism

Important part of our solution is mutual communication among multiple mobile devices – OwNet nodes. An OwNet node can communicate with other nodes within the same Wi-Fi network using our own communication protocol. This communication is used to share metadata related to cached content to ensure that no content is unnecessarily downloaded multiple times. Instead of re-downloading it from the Internet, an OwNet node can request the content from other node on the local network. This is especially useful for viral content or content related to a current lesson at school (which is downloaded once and quickly distributed to others).

The main principle of our protocol is that all peers identify one master node among them which acts as router, holding complete metadata of the distributed cache.

3 Conclusions

We have described our work on the *OwNet for Android* project. Our solution targets Android-based devices as we believe that mobile computing is going to prevail in developing countries such as in Kenya (where almost everyone has a mobile phone, while only a fraction of population has an access to a computer) and that open-sourced Android will be the mobile platform with most of impact in these countries.

Our application acts like a proxy server installed on a localhost, capturing web requests from a web browser and using its cache to serve responses. The real advantage of Ownet emerges when a group of people start to surf on the Web via OwNet within the same local network. Distributed cache along with collaborative features of bundled web portal makes surfing faster, less sensitive for connection dropouts and more social – if one member of a group finds an interesting resource, it is immediately available to others, without posing additional burden on an Internet connection, which can be meanwhile used to fetch some other content.

References

[1] Demovič, Ľ., et al.: Enhancing Web Surfing Experience in Conditions of Slow And Intermittent Internet Connection. In: *Information Science and Technologies Bulletin of ACM Slovakia*, Vol. 4, No. 2, (2012), pp. 25–29.

OwNet: Your Own Personal Internet

Karol BALKO*, Michal DORNER†, Martin KONÔPKA†, Marek LÁNI†,
Martin LIPTÁK†, Andrea ŠTEŇOVÁ†, Matúš TOMLEIN†

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
12tim2012@gmail.com

1 Idea

The Web is inarguably an unlimited source of information. Even though it has become essential part of our lives, it is still not accessible to everyone. There are many barriers to reach fast and reliable Internet connection in developing countries. Moreover these barriers are often present in developed countries as well. The area that can benefit from good-quality Internet connection the most is education. However, there are many schools where several students share slow and intermittent connectivity. These issues will be solved with better infrastructure in the future, but a good Internet connection is needed now.

We are developing a pure software solution called OwNet. The first part of OwNet is a *local proxy server* that utilizes available Internet connection using personalized *caching* and *prefetching* of web content. Cached or prefetched web content can be accessed offline. The second part provides tools for *content sharing* and *collaboration* that are accessible even offline. OwNet clients communicate with each other on local network to share their cache and other data.

A small bar is injected to every page to access content sharing and collaboration features. Users can share a web page or post a message to all the other users or privately in user groups. Users can access intranet portal to find activities, shared content and materials posted by other users. Links accessible offline are highlighted so that users can browse without seeing the error page about being disconnected. When users attempt to visit a page that has not been cached offline, they can schedule its download later when they get back online.

2 OwNet v1: Lessons Learned

Part of our team members worked on OwNet one year ago as a part of international competition Imagine Cup [1]. The original OwNet was implemented using single-platform technologies. It used client-server architecture for local network cache and data sharing. We managed to deploy it in schools in Kenya, which gave us valuable lessons. Original client-server architecture required complicated setup and a powerful computer that acted as a local server. Computer labs in developing countries have no skilled administrators to configure the local server and there are old computers that do not meet requirements for this kind of architecture. There is also high chance of

---

* Master degree study programme in field: Information Systems

† Master degree study programme in field: Software Engineering
Supervisor: Karol Rástočný, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

this single local server going offline and disabling other users. We decided to apply easier to setup and less error-prone peer-to-peer architecture, make the application multi-platform and to use re-implementation for other technical and user experience improvements.

There is no more need for manual selection of local server. All instances on local network are simply called clients. However, the most powerful computer is chosen as a master client. This selection is done seamlessly by *multicast protocol* we have developed. This computer is used as a central point when other clients synchronize their local data. It is much more efficient than a truly peer-to-peer synchronization (every client synchronizes with every other client) when communication overhead grows exponentially [2].

OwNet stores metadata about caches and other data about user activities, messages, recommendations and shared content in a relational database. This database is synchronized across all clients on the local network. Web content caches are not synchronized, as they can grow large and they can be downloaded again in case of their inaccessibility (when computer containing requested cache files is shut down). Only cache metadata are synchronized so that every computer knows where to find requested cache files on the local network. Besides lack of a single point of failure, advantage of this approach is accessibility of all data (except full non-local caches) offline, for example when student is at home.

Developing a multi-platform desktop application is a non-trivial task. Developers have to solve many platform-specific problems and bugs. We try to prevent bugs using automated testing and modular architecture. We need to take care of things like service monitoring and error reporting. Also *easy installation* and *automatic updates* are essential parts of our solution. *Modular architecture* allows us to add modules with new functionality or to replace modules in current installation in much easier way than it could be done before. Even though dramatical architectural and technology changes have required us to completely re-implement OwNet, many issues present in the original version are solved (e.g., highly coupled internal components or complicated update process of installed versions).

3 Deployment

We cooperate with two Slovak NGOs: People In Peril and Pontis. They successfully deployed original OwNet in 2 schools in regions of Kenya. Unfortunately, we are unaware of it being used by students. Nevertheless we got positive feedback from teachers and their enthusiasm to use OwNet. Together with People In Peril and Kenyan company Astor Computer Solutions we were successful with our application for Slovak Aid small grant to connect 6 schools in other regions of Kenya to the Internet and deploy OwNet. Pontis organization offered us help with deploying OwNet in other 5 schools in Kenya in March 2013. Moreover we have acquired contacts to schools in Guatemala, where they also suffer from poor Internet connection and may benefit from the OwNet solution.

References

[1] Demovič, Ľ., Konôpka, M., Láni, M., Tomlein, M.: Enhancing Web Surfing Experience in Conditions of Slow and Intermittent Internet Connection. *Information Sciences and Technologies. Bulletin of the ACM Slovakia*, (2012), vol. 4, no. 2. pp. 25–29.

[2] Iyer, S., Rowstron, A., Druschel, P.: Squirrel: A Decentralized Peer-to-peer Web Cache. In: *Proc. of the 21st annual symposium on Principles of distributed computing (PODC '02)*. ACM Press, (2002), pp. 213–222.

Emotional State Recognition

Michal BIROŠ**, Tomáš CABAN*, Tomáš KUNKA*, Filip STAŇO*,
Tomáš LEKEŇ*, Milan MARTINKOVIČ*, Bálint SZILVA**

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
teamnumero14@googlegroups.com

1 Introduction

Unlike machines, human performance is definitely affected by feelings and emotions. Each computer user experiences different emotions. If the user successfully completes project, his emotions are most likely going to be positive. On the other hand, if he loses important data due to system failure, negative emotions are going to be dominant in his facial expression. These emotions in connection with activities which cause them are good source of information and can provide feedback needed in many business sectors. For example, monitoring and keeping programmers in positive and productive mood may lead to more than good results.

Emotion Log is watching application constantly and periodically capturing user's face and recognizing his actual emotional state. We are working with points of user face represented by x and y coordinates. We have developed method which uses simple computation based on position of points and we also use machine learning techniques.

Emotion Log is extension of running project *PerConIK*. *PerConIK* is application capturing information like keystrokes or open tabs in browser. We extended this project with part acquiring video data from camera. Our goal is to use this data to correctly recognize human emotions from users face expression almost in real time.

2 Emotion Log Design

The core functionality of *Emotion Log* is based on the use of extern library *Luxand*. It provides basic operations for working with images and video. Using this library we were able to capture user's face each second. Each second we capture and extract sixty-six points of user face. We are working with 2D images therefore we represent each point with x and y coordinate. This data alongside other gathered by *PerConIK* is stored in local database on user computer. Periodically every 30 seconds the data is gathered and sent from user's local database to server.

2.1 Emotion recognition approaches

Emotion recognition is based on three different approaches:

- Emotion recognition from neutral state

* Master degree study programme in field: Software Engineering
** Master degree study programme in field: Information Systems
Supervisor: Martin Labaj, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

- Library for Support Vector Machines
- Neural network

Emotion recognition from neutral state is based on assumption, that neutral state is the most common state of face. Therefore the neutral state algorithm observes user for a short period of time and then determines neutral state based on frequency of similar outputs.

Every personal profile is required to have neutral state coordinates. Recognition of emotion is based on comparing neutral distances of two important points of face from each other, such as distance of inner eyebrow point from inner eye point, with actual state of face values. Such differences in these values may indicate different emotional states: happiness (smile), surprise (eyebrows distance from eye grows), anger, sadness, exhaustion.

Figure 1. Demonstration of emotion recognition from neutral state.

LibSVM is a library for Support Vector Machine. This supervised learning model is another approach of emotion recognition based on actual differences between coordinate indicators of face points. Support Vector Machine is not using neutral state for comparison as it is only trying to learn specific values for every emotion for all test subjects.

Neural network is a machine learning approach inspired by biological neural networks. We use this approach to recognize user's emotional state. Along with LibSVM, neural network uses training set Bosphorus [1] of people's faces to learn facial differences between emotions.

Value of emotional state grows if the emotion is expressed for a longer period. We are setting score for each sequence of emotions, which assures finding proper moment to create recommendation. With this approach we are trying to avoid creating recommendations after small, short-term fluctuations in facial expression of user.

3 Conclusion

We have been able to recognise actual state of emotion. Application is extensible, so new emotion recognition approaches can be easily build in. Second big portion of our work apart from emotion recognition will be appropriate recommendation for user of *Emotion Log*, like recommending little break for tired or stressed user.

References

[1] A. Savran, B. Sankur, M. T. Bilge: Comparative Evaluation of 3D versus 2D Modality for Automatic Detection of Facial Action Units, *Pattern Recognition*, Vol. 45, Issue 2, 2012, pp. 767–782.

Intuitive Control of Multimedia Home

Ivana BOHUNICKÁ[*], Ján GREPPEL[*], Juraj MURÁNSKY[†], František NAGY[*],
Dominik RERKO[*], Matúš UJHELYI[*], Zuzana UJHELYIOVÁ[†]

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
timak_tim11@googlegroups.com

Current trend in modern society is bringing information technology into places and situation that the first computer makers could not imagine. It is common for household to own several laptops, smart phones, televisions or other devices [1]. Every day we meet with more capable and compact devices, which are able to perform various actions. For this reason, there is increasing pressure on system creators to make more intuitive interfaces providing the easiest user interaction. However, using the interfaces there is often natural limitation either by technological details or by interface of given device. Users with some kind of disability or elderly people can have difficulties with learning to work with modern technologies.

It would be more comfortable for the user if the device was able to respond to custom commands, which the user defined. Anything can be considered an input: a gesture, motion, command, speech, smart phone gesture or a key press on an infrared controller. New consumer devices, such as the Kinect sensors, allow us to work with this kind of input data. Kinect is able to recognize visual and acoustic information from the environment. The Kinect Software Development Kit (Kinect SDK) gives access to skeletal tracking using an intuitive API [2]. This information can be used to capture and recognize gestures trained by the user.

We designed and developed a complex system that enables user to control any supported device with any input event. At the same time we allow a user to control multiple devices at once or to define same input event for multiple actions to multiple devices. The system is also supposed to be flexible enough, so it would be possible to simply add more devices to it (Figure 1).

The main goal of the project is creating highly modular software system running on a single personal computer, to which could connect client applications over standard networks. The major element would be a central application able to on the one hand to accept variety of inputs, for example in form of gestures or voice commands from Kinect sensor. On the other hand it would be able to notify and communicate with other applications over standard network protocols or messaging systems. This module structure provides options for simple future extension with other modules, which would actually be only new applications connecting to the main central application, also called central module.

In the first phase of solution, we managed to design and implement a prototype of modular system communicating with the external devices over the wireless network. We designed and implemented a custom communication protocol over TCP for the communication of devices with

[*] Master degree study programme in field: Software Engineering
[†] Master degree study programme in field: Information Systems
Supervisor: Michal Kottman, Institute of Applied Informatics, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 455–456.

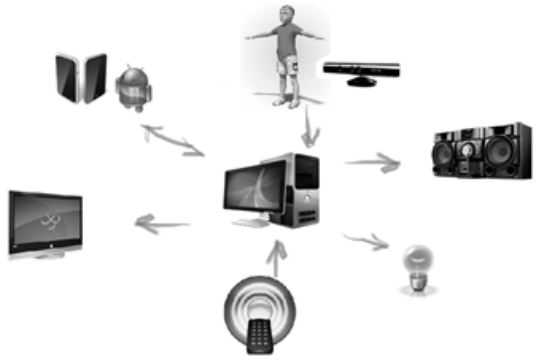


Figure 1. Communication of the modules.

the central module over the network. A sending device is any device that can send input events to the system and a receiving device is any device that can receive events and execute actions. During connection, the sending device informs the central library of its supported input events, and a receiving device informs of actions it can execute.

In our solution the sending devices can be sensor Kinect, which can recognize movement, mobile phone, which can recognize gestures on display or human voice and infra-red TV remote controller. Receiving devices can be multimedia equipment such as TV, Hi-fi or mobile phone, on which it is possible to perform actions such as turning on the LED diode or playing an audio.

Based on the set of events and actions, it is possible to define a situation. Situation is a pair event → action that defines which action should be executed when the particular event occurs. It is possible to map multiple actions to the single event, so in example the wave in front of the Kinect sensor can simultaneously turn off the television and turn on the LED diode on the mobile phone. To create an option of realization multiple situations simultaneously, we created modes. Mode represents a sort of state in which the central module is. There are two types of modes, local and global. It is possible to have several local modes, but there is only one global mode. The user can switch between local modes. The situations from the global mode are accessible all the time. In each mode, there can be mapped several situations. Based on the mapped situations, it is possible to create modes for the living room or to turn off the sounds in whole house. Main goal is simplification of control of devices for users.

We designed the modular multimedia system able to communicate with input and output devices over the network. The goal is to enable control of the multimedia devices in household through different devices, using even intuitive inputs, such as voice or motion. Another goal is to improve existing solution and also increase the set of devices, which are able to be connected and controlled from the central application, such as Raspberry Pi, BeagleBone and light bulb.

References

[1] Mrazovac, B.; Bjelica, M.Z.; Papp, I.; Teslic, N.: Smart Audio/Video Playback Control Based on Presence Detection and User Localization in Home Environment, Engineering of Computer Based Systems (ECBS-EERC), *2011 2nd Eastern European Regional Conference on the*, vol., pp. 44–53.

[2] Yi Li: Hand gesture recognition using Kinect, Software Engineering and Service Science (ICSESS), *2012 IEEE 3rd International Conference on*, pp. 196–199.

Televido – My Personalized TV

Ľuboš DEMOVIČ, Eduard FRITSCHER, Jakub KŘÍŽ
Ondrej KUZMÍK, Ondrej PROKSA, Diana VANDLÍKOVÁ*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
televido@googlegroups.com

In today's world there is a large amout of media content produced every day: movies, TV shows, TV programs etc. With so many options, chosing the right content can be difficult and overwhelming for the user. Take television. There are tens of channels from which the user can chose at any given moment. If he choses the wrong channel, he might miss something he would enjoy on a different one. The problem becomes even larger with more and more popular television or video on-demand. The user has literally milions of possibilities from which to choose. The task becomes very difficult without a personalized recommendation system.

Recommendation systems are systems which can analyse the taste, the mood or the context in which the user is at the moment. Based on the analysis, they create an accurate recommendation that suits the particular user [2]. The two main categories of recommendation systems are content based and collaborative. In both categories the algorithm has to go through the entire entity base to find the correct content to be recommended. There is no guarantee that the estimation is correct and the recommended content is accurate enough for the user. Many recommendation systems try to recommend content by pairing the extracted knowledge base with the user's context and taste. Using relational databases these tasks can be difficult and not very efficient. As a result, the recommendation systems may suffer from performance issues and, to be able to use them in real time, the recommendations need to be cached or the entity base simplified.

We aim to design and evaluate a recommendation method that uses a new approach for recommending multimedia content. Instead of the approaches mentioned above we decided to design a service which uses a hybrid model of relational and graph databases for recommending multimedia content, in particular movies, TV shows and TV programs. Graph-based recommender systems have been tested in the past and have shown promising results [1].

The design of the architecture we use is shown in Figure 1. As mentioned, we use two types of databases – relational and graph database. Relational databases are widely used for storing basic data. We use it to store basic information about entities, such as title, description or creation date. This information is easily and quickly fetched for the user interface and can be used to perform faceted searches.

Relational databases are inefficient for storing graph structures, which is the reason why we make use of a graph database. The graph database contains the domain model used for recommendations. The model is based on relations between entities – the main entities are movies, shows and programs.

---

* Master study programme in field: Software Engineering
Supervisor: Dušan Zeleník, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 457–458.

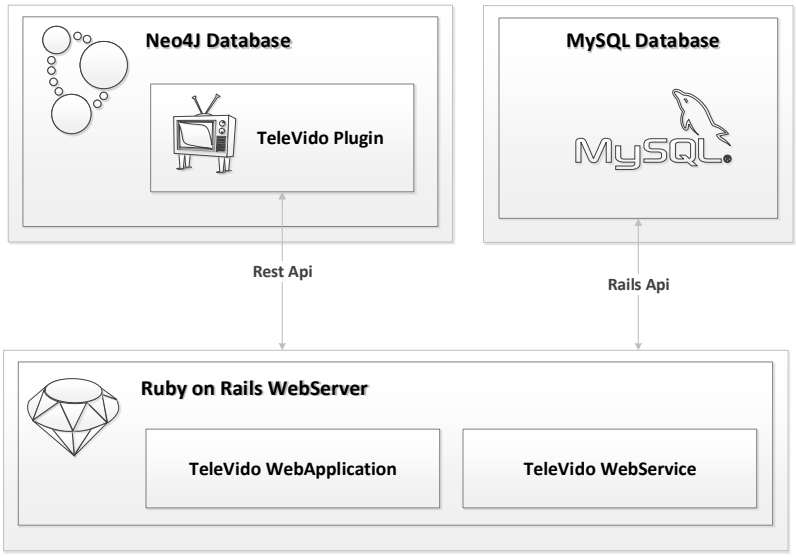


Figure 1. Televido architecture.

These are connected to each other via relationships, which are based on other entities like actors, producers, directors and genres.

The recommendation algorithms are created as plug-ins to the graph database. They are graph algorithms, which work from initial nodes. The initial nodes are currently selected explicitly, but we plan on selecting them implicitly based on the user model in the future. The algorithms try to find nodes which are closest in the graph to all initial nodes, which are then returned as recommendations. There are multiple ways of looking at the problem of finding the closest nodes, especially in a very complex graph, which is why we designed and implemented four separate algorithms. These algoritms were designed to take in multiple parameters, such as subset of nodes which can be returned as recommendations or subset of relationships which can be used to crawl the graph. Thanks to these parameters, the recommendation algorithms are quite agile. For example, the same algorithm with different parameters can be used to recommend movies which are currently being shown at the cinemas or only the TV programs which will be on tommorow night.

Our recommendation algorithms are currently being experimentaly evaluated. Based on the results of the experiments we will be able to determine the accuracy of each algorithm, select the best algorithm and further tweak it to perfection.

The main advantage of using a graph database is the performance. With enhanced graph algorithms we do not need caching of the recommended content because the service is fast enough to work in real time.

References

[1] Huang, Z., Chung, W., Ong, T.H., Chen, H.: A graph-based recommender system for digital library. In: *Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries*. JCDL '02, New York, NY, USA, ACM, 2002, pp. 65–73.

[2] Ricci, F., Rokach, L., Shapira, B.: Introduction to Recommender Systems Handbook. In Ricci, F., Rokach, L., Shapira, B., Kantor, P.B., eds.: *Recommender Systems Handbook*. Springer US, 2011, pp. 1–35.

Crowdsourcing Pictures of Real-World Places

Peter DULAČKA*, Tomáš FILČÁK*, Michal LIHOCKÝ*,
Lukáš ĽOCH*, Matúš MICHALKO*, Marek ŠUREK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
gangstacoders@gmail.com

People often visit renowned places and many tend to take pictures during their travels. These places may be famous as well as unknown, which influences amount of information about the place available on the Web. New generation of smartphones is now widespread across population and it brings possibility not only to use the cell phone as a camera, but to use it as a device for sharing information real-time. Considering existing guidelines [1] for creating location based game (LBG), we aim to reach two goals:

- To acquire pictures of places suffering from lack of available public photo-documentation.
- To make people visit new places and meet new people.

To achieve our goals we propose a mobile location-based GPS game called *"The Cartel"*. The main goal of a player is to play given mission and grab a reward. There are two kinds of missions: 1) *storyline (primary) missions* (as a fun support and motivation for players) and 2) *side (secondary) missions* (with objectives stated openly, which forces the player to explore the unknown in order to complete the mission).

The most motivating feature of our game is its story line (Figure 1e). Since the game is location-based, it is not possible to create storyline globally – mission spots have to be selected manually; hence we decided to limit the story for the city of Bratislava. We have picked numerous places which lack public photo-documentation or fit into the storyline and placed game missions near them. Player starts as an associate in a mob organization. By visiting places and completing missions (Figure 1c) he struggles his way to the top of *"The Family"*. After the player gets to the place of a mission, he receives further information from his contact (Figure 1a). Missions might consist of objectives such as pursuing fake person, inspection of chosen place or escaping from the place within limited time. Each of these objectives may be *terminated by taking a picture of place* (e.g. player has to walk around a building and take a shot from all four angles). These are later being uploaded to public gallery containing crowdsourced images from all players.

When the player's ranking gets higher, he can join forces with other players and run their own mob racket. The ranks in the game are designed to be close to ranks in real organization (Figure 4e). By completing secondary missions (such as visiting three restaurants with at least two other people) player expands the sphere of his racket and gets a control over the city. To maintain the control over the racket the members of group have to explore new places in the area of their

* Master degree study programme in field: Software Engineering
Supervisor: Jakub Šimko, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 459–460.

governance. By doing this, people are continuously meeting new people (the bigger the group, the higher profit for the racket) and exploring places they may have never been to.

Reward for the completed story missions and rackets is a virtual currency that can be used for buying or trading items which help the player solve future missions – such as a tip for exploring or item needed for mission. This creates constant need for maintaining player's racket and game continuality, yet still differs from games such as *"Big Game Hunter"* [2] where players have to create missions for other players which might not be much fancied.

Figure 1. Screenshots of the mobile application "The Cartel".

To maintain a public gallery of crowdsourced images, we realized a website that communicates with the mobile game and that shows all public data generated by the players. The game is designed to reward players for submitting their photographs to the game's public gallery. However, players may cheat in order to receive a reward and therefore only small portion of the reward is given to the player right after he submits the picture. The rest of the reward (i.e. the bigger part) is given to him after confirmation of originality of picture (so it cannot be downloaded from the Web or a picture of completely different place than stated). This verification is being done not only by administrators but also by other players (who are being rewarded for doing so) directly in one of the game modes.

The game has two (client-server) parts. Mobile part is based on Android platform and in the current state it requires a device with a camera, GPS locator and Internet connectivity (satellite map is mandatory for missions). Harvested data are being stored locally on the device until Wi-Fi connection is available; then they are synced with server. Although the continuous data harvesting might be turned off (due to privacy issues), the tracking of position while the game is active is mandatory (e.g. GPS location of place where the picture had been taken).

The game is still in its development stage and some of the mentioned features are just yet to be implemented. Globally we would like to integrate the story even more into the real life – mostly by picking "the right" spots for the missions and by better integration of areas affected by rackets. In case the game is successful, switching to player-generated content might be necessary.

References

[1] Neustaedter, C., Tang, A., Judge, T. K.: Creating scalable location-based games: lessons from Geocaching. *Personal and Ubiquitous Computing*, vol. 17, no. 2, pp. 335–349, 2011.

[2] Lochrie, M., Lund, K., Coulton, P.: Community generated location based gaming. *Proceedings of the 24th BCS Interaction Specialist Group Conference*, pp. 474–478, 2010.

Detecting User's Emotional State

Samo FORUS†, Jozef GAJDOŠ*, Martin GEIER*, Peter GREGUŠ*,
Miroslav HUDÁK*, Peter SIVÁK*, Peter ŠINSKÝ†

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
team.10.tp@gmail.com

Emotional state is a factor which greatly affects user's productivity when working with a computer. It may change many times during the work and sometimes the current mood of a user is not appropriate for best results. Moreover, the user does not have to know his actual emotional state.

The aim of this project is to create software capable of monitoring user's activity during his work with a computer and according to that detect his emotional state and optionally make some recommendation to the user which will increase his productivity at work. In this project we detect user's emotions by monitoring only his work with a keyboard and a mouse. This is an advantage because these devices are both inexpensive and commonly available and they do not annoy the user as opposed to other devices measuring biometric information like pulse rate or EEG.

The first part of the whole recognition process is logging of user activities. We log individual keystrokes, mouse button presses and mouse moves in time and during this process the user manually specifies his current emotional state. After sufficient amount of tracked samples we get a model of the user for the individual emotions and after that we can ask user for his emotional state less often to update his model. Every user has to have its own model because different users behave differently at the same emotional state [1]. With several detecting algorithms we can find out an estimated emotional state from tracked samples without manually retrieving this state from the user. According to the obtained emotional state we can provide the user a recommendation.

The whole process of our solution can be broken down into four basic blocks – user monitoring, user modeling, model recognition and recommendation providing.

We have designed our solution such that we can monitor multiple users at the same time by dividing our application into two parts – client and server part which is shown in Figure 1. Each user installed the program *PerConIK* [3] on his local computer which monitors his activity and according to this activity the user is "modeled". After specific intervals of time the user models from different computers are sent to the central server where these models are recognized and current emotions of users are found out. When multiple users send multiple models at the same time, the models are pushed to front and they wait for processing. According to calculated emotional states recommendations are created, which are sent back to individual local computers.

We have successfully completed a functional prototype which is able to perform basic tasks defined in project goals. We have extended the project *PerConIK* capable of monitoring user's

---

* Master degree study programme in field: Software Engineering

† Master degree study programme in field: Information Systems
Supervisor: Assoc. Professor Daniela Chudá, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 461–462.

activities to be able to ask a user for his current emotional state during selected time intervals and to send this information with corresponding activities to server where they are stored to a database.

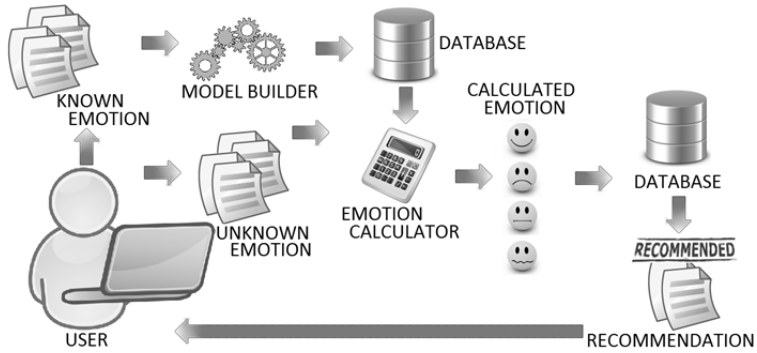


Figure 1. User's emotional state detection cycle.

We have designed a user friendly interface for selecting a current emotional state which should not distract a user from his work. We have designed three algorithms [2] for comparing user's model and emotional vector where two of them were relatively successful. We have also designed a mechanism for committing and choosing a recommendation. All these designs have been properly implemented and experimentally verified.

In Table 1 is shown one of the comparing methods which gave us the following results when minimum similarity has been set.

Table 1. Comparing method with best results.

Emotion	True	False	Count	Count of activity	Weight
Normal	0,768559	0,231441	229	8400	176
Happiness	1	0	5	589	5
Anger	0,428571	0,571429	7	120	3
Disgust	0,666667	0,333333	9	526	6
Tired	0,888889	0,111111	9	2441	8
Stressed	0,666667	0,333333	6	181	4
Nervous	0,666667	0,333333	9	119	6
Accuracy					75,91241

Emotional state represents a significant factor which affects our total productivity at work. Correct identification of current emotional state and subsequently appropriate recommendation are key factors in increasing this productivity. We believe that this project can solve this problem and all its corresponding sub problems – user monitoring and modeling, model recognition and providing an appropriate recommendation.

References

[1] Epp, C., Lippold, M., Serpette, Mandryk, R.L.: *Identifying Emotional States using Keystroke Dynamics*. CHI, 2011.

[2] Brown, M., Rogers S. J.: *User identification via keystroke characteristics of typed names using neural networks*. Machine Studies, 1993, pp. 999-1014.

[3] Gratex International a.s. PerConIK. Available at: http://perconik.fiit.stuba.sk/. 2013.

Recommender System for Multimedia Content

Michal GRANEC*, Tomáš JENDEK*, Ján KANDRÁČ*, Ondrej KAŠŠÁK*,
Ján TREBUĽA*, Maroš URBANČOK*, Juraj VIŠŇOVSKÝ*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
tp-tim04@gmail.com

Nowadays we are facing the problem of the information overload, especially in the multimedia domain. As the amount of data keep growing, users have access to large number of multimedia contents and face problem of processing variety of information. Even choosing what to watch in TV becomes more complicated due to a wide range of options and huge amount of multimedia content available. User desires to spend his time effectively and find relevant information in the shortest time possible. Therefore there is a demand for personalized multimedia content recommendation.

Our solution is designed for people who like to spend their leisure time watching multimedia such as films or TV shows. These people are facing problem with processing huge amount of information in order to find appropriate content and we want to make it easier for them. Our aim is to provide personalized recommendation, effective searching and filtering tools. To achieve this we developed Loomie TV, a social oriented web application for recommendation of multimedia content. Benefits of social networking connected with multimedia domain brings us new possibilities for better recommendation and building stable user community.

Filtering information is the usual way, how to face up the information overload problem. In our solution we push it forward by designing so-called turbo filter. Thanks to simple and intuitive graphic interface, users are able to construct complex queries easily in order to access desired multimedia content.

There is problem with obtaining appropriate metadata for multimedia because of difficulty and high computational complexity of video or audio analyzing, which restricts filtering options. Therefore simple filter can not be considered a sufficient solution. Because of these problems there is a demand to solve the multimedia overload problem by personalized recommendation. In our work we use hybrid approach to fulfil recommendation tasks [2]. Content-based component of this recommendation approach is meant to recommend items depending on user's past viewing activities and thus, it corresponds to his interests. The set of recommended items is then extended by items liked by similar users. This extension allows user to get to know items which does not really correspond to his viewing preferences, but he is assumed to like them as these items were approved by similar users.

In order to provide sufficient personalized recommendations for every user, we process his activity (movie ratings, reviews written). Based on this information, the user model is constructed and in the next phase personalized recommendations are generated.

---

* Master degree study programme in field: Software Engineering
Supervisor: Michal Kompan, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 463–464.

We distinguish following types of relationships between users within Loomie TV portal-friendships, following authorities and similar users. Friendship is a relationship between two users of equal status. Friends can influence each other by their activities. Posting reviews about the movies they have recently watched, marking movies they wish to see in the near future, or assigning ratings to movies they like or dislike may influence their friends selection of movies to watch.

Following, vice versa, is a one-sided relationship, where one user considers another as an authority and is interested in his activity. The follower is therefore notified about assigned reviews and written reviews by the authorities he follows. On the other hand, the authority has no interest to be aware of activity realized by his followers.

The third type of relationship is similarity between two users [1]. The degree of similarity is defined by their shared interests or similar ratings of multimedia content. Thanks to discovery of similar users we are able to enrich personalized recommendations based on user's viewing history, by items liked by the most similar users.

Watching TV or multimedia in general is considered to be a group activity, as many people do not watch multimedia all by themselves. For viewing group is characteristic its changeability, because in most of the sessions group consists of different set of members. Therefore it is necessary to build model of users for each session [3]. Group model is based on viewing preferences of every group member. The main purpose of this model is to generate recommendations to this group. Yet it is necessary to bear in mind that it is desired to satisfy every single member of the group. Unsatisfied user may consider avoiding the use of our system in the future and he may discourage other members of the group as well. Therefore, we consider ensuring each group member’s satisfaction an important aspect of the quality of recommendation.

Our solution supports organization of watching events in groups. This support is provided by social network in our system, where users can easily arrange watching events and subsequently evaluate what they have seen. Based on this after-watching evaluation we are able to improve user model of group members. After-watching feedback collection is realized by an application for mobile devices. The application extends common ways of feedback collection, as it allows group members or a particular user to rate viewed item immediately after watching.

Another aim of our work is to offer recommendations to the user as soon as possible. One of the possible solutions is to ask the user a series of short questions. Questions are invariable and together with answers they compose a decision tree, where questions represent nodes and answers are edges to the children. By answering this set of questions, users define his path in the tree. After answering all of the questions user is classified into a group and is offered results regarding to the group model. It is clear that the questions in the tree must be ordered, so that the most restrictive ones are placed as high as possible in the tree, so that they are asked first.

References

[1] Drago, I.: A Multi-measure Nearest Neighbor Algorithm. In: *11th Ibero-American Conference on AI*, Lisbon, Portugal, Springer-Verlag, 2008, pp. 153–162.

[2] Ricci, F., Rokach, L., Shapira, B., et al.: Recommender Systems Handbook, MA: Springer US, 2011, pp. 73–105.

[3] Senot, C., Kostadinov, D., Bouzid, M., et al.: Analysis of Strategies for Building Group Profiles. In: *User Modeling Adaptation and Personalization*. Vol. 6075. Springer Berlin. 2010, pp. 40–51.

Innovative Mobile Game Focused on Environmental Issues

Gabriel MANČÍK*, Šimon MIKUDA*, Juraj PITÁK*, Róbert PUCKALLÉR*,
Michal RAČKO*, Jozef REŠETÁR*, Bohuš ROŠKO*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
tp_tim_5@googlegroups.com

The aim of this project is to create an innovative computer game and motivate children and adolescents to use leisure time meaningfully and in addition to that have fun. Such a game should be interesting, intuitive and last but not least educational. It should motivate the target audience and promote their desire to acquire new information. So we decided that the most appropriate tool would be game for mobile devices that modern children are familiar with.

Our aim was that game will be designed not only to have educational character, but also to be fun. We used already known methods for creating games and put into it our own elements. With these devices are children used to work, or better said, have fun, so for them it will not be a new experience.

The game is composed of two parts that work together. The first part, river, is located in the lower part of the game screen, in which the garbage is flowing. The role of the player is to get them into the factory over the river. These factories are able to produce ammunition for towers. Here comes into play the second part, in which the player must defend against enemies. Those enemies come in lines from above the towers, trying to destroy factories and empty the waste bag that are they carrying back into the river. Player's task is to destroy them with weapons produced in factory.

Factories are divided into seven categories according to the type of waste that they can handle. Those types are wood, glass, paper, rubber, oil, metal and plastic. The types of ammunition made from different type of waste work with varying effectiveness against different types of enemies. Towers can move on a track from one line to another, and thus take advantage against the enemy. Some enemies are heavily armoured and it is necessary to combine the power of two towers. Therefore, we introduced to the game unique element. Player is able by pulling one tower over the other combine their effects and thus stop stronger enemies.

The essence of the game is to prevent pollution of the river. This river must be kept clean to avoid overfilling river with waste. At the end of the river is wastewater treatment plant that has limited capacity. Therefore, when large quantities of waste are present in river, it will fill and the player loses. Also enemies who come to the river are carrying bags of waste that will increase pollution. Those bags are necessary to torn apart and then sort it into the correct factories. As mentioned above, the enemies are strong and with different types of armour worn by many.

---

* Master degree study programme in field: Software Engineering/Information Systems
Supervisor: Eduard Kuric, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 465–466.

Information about the strength and durability of every enemy player appear at the beginning of level in order to acknowledge player what threat he faces.

An equally important objective for the players, in addition to entertainment, is to learn about the importance of waste separation. This is emphasized not only in the game already polluted river, but also enemies who are trying to continue to pollute it. Our game was created in order to think about this fact and help clean up the river at least virtually.

The individual levels are designed and constructed with the increasing complexity and in each set new enemy is introduced. In the later stages of the game the enemies are combined and player must apply his knowledge to defend himself. Levels can be easily created using the XML description, which stores all the text that is displayed to player. Also it contains description of game elements, their abilities, strength and timing of enemies coming in lanes. Waste is spawned randomly with increasing rate and difficulty to separate.

Figure 1. Game prototype.

Our goal was to design and build a prototype of an innovative computer game. The resulting prototype for Windows Phone 7 mobile platform has been internally tested. We have simulated and enhanced interaction with individual elements to our needs and best game experience. In the current state of the project, we focused on the area of game mechanics and waste flow simulation. Another objective was to compel the player to think about nature and pollution of water resources in the form of entertaining game. It combines two unique game mechanics to make the game a challenging for the player.

In the future we plan enrich the game with several more types of enemies and enhance the graphical aspects of the game. Also we want to add sound effects into the game, which would magnify the whole gaming experience. The game for the purpose of testing features a limited number of levels needed to test each type of game element. However, levels do not provide any method of difficulty progression or logical sequence of enemies. Therefore we are planning to create more consecutive levels to immerse players into the game and convince him that he wants to continue playing.

References

[1] Schell, J.: The Art of Game Design: A book of lenses. *Annals of Physics,* Vol. 54, Morgan Kaufmann, (2008), p. 489.

[2] Ding, S.: The Reasonable Combination of Game and Teaching. *2011 International Conference on Control Automation and Systems Engineering CASE 1–3*, IEEE, (2011).

RoboCup Presentation at IIT.SRC 2013

Pavol NÁVRAT, Ivan KAPUSTÍK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
{navrat,kapustik}@fiit.stuba.sk

Abstract. RoboCup is an attractive project theme with a free participation, designed to support education and research in artificial intelligence, robotics and information technologies. During the last few years, our students achieved some interesting results, which were presented during our student research conference.

1 Motivation

RoboCup is an international joint project to promote research in artificial intelligence, robotics and information technologies. It is an attempt to advance artificial intelligence and intelligent robotics study and research by providing a popular problem where wide range of technologies can be integrated and examined. RoboCup chose to use soccer game as a central topic of research. The ultimate goal of the RoboCup project is to develop by 2050 a team of fully autonomous humanoid robots that can win against the current human world champion team in soccer.

In order for a robot team to actually perform a soccer game, various technologies must be incorporated, including design principles of autonomous agents, multi-agent collaboration, strategy acquisition, real-time reasoning, multi-level decision making, robotics and sensor-fusion. RoboCup is a task for a team of multiple fast-moving and skilled robots within a dynamic environment. It offers also a software platform for research on the software aspects. RoboCup is divided into four main fields: RoboCup Soccer – defined by the original domain of soccer, RoboCup Rescue – intended to do search and rescue in large scale disaster area, RoboCup Junior – aimed to child education and motivation and RoboCup @Home – oriented to provide various help not only at home.

From our point of view, the main goal of RoboCup is to promote research in areas of artificial intelligence and information technologies, especially in the area of multi-agent systems. This is a benefit for the students, making their studies more interesting and attractive. Students can meet with robotic soccer in courses like Artificial Intelligence, Team Project and others. Students are facing an interesting problem, which demands invention as well as use of modern artificial intelligence approaches. Teams of students have the possibility to directly compare their results in tournaments. This encourages the students to even higher effort and motivates them for better results. More fundamentally, achieving progress requires tackling serious open research problems in artificial intelligence, such as planning of cooperation of multiple agents etc. That is why this area is of interest also for our doctorate students.

* Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

We have been organizing this tournament regularly for several years. Starting as a local event in 2000, it has grown to a regional contest under the official RoboCup authorization. Our Faculty organizes tournaments in the simulated category only, but we gradually include other categories. Our current contest event has three parts.

First part is an exhibition of two-dimensional (2D) simulated player teams, where students try to make their own players win soccer game. 2D players are simple entities, ready to follow any possible action in their virtual environment. Students' main research is aimed to team tactics and autonomous player decision. It covers team formations and planning, player communication, and use of a team coach. Recent projects are oriented on decision skill improvement. Methods here cover planning and player's action selection based on diverse sources – success evaluation of similar situation, teammate decision model and prediction of opponent behaviour. This contest part is more an exhibition of new approaches than a tournament, because our interest shifted to more complex three-dimensional (3D) robotic simulation.

Second and third parts of this event involve the 3D simulation. These robots are true copies of their real master. They have limbs and joints and their only action is to turn chosen joint. Primary students' task was to teach robots to reliably walk, turn, stand up and kick the ball. It was followed by design of a proper composition of these basic skills to achieve simple goals, like walking to the best game position or getting the ball. Then, the training support framework has been developed and test modules for robot learning were created. Currently, students train robots in many ways: improvement of basic skills, dynamic balancing, obstacle detection and smart avoidance, automatic situation recognizing and the best action selection to reach the main goal – win the soccer game.

Any soccer player must be good with physical skills and must make good and fast decisions during the game. So the second part of our event contains skills competition. Robots compete in speed and accuracy in performing given tasks. They can get a few points for "unusual" useful skills as well. Finally, third and most valued part of this tournament holds soccer contest, where both skills and decision making are tested in real-time game.

2 Results presentation

For this student conference we decided to hold an exhibition of results achieved in 3D soccer simulation. Few student groups work on new skills and decision making for 3D soccer robotic players. Every group presented details about their own ideas and methods. These methods involve new movement and skill design, optimal robot trajectory building, structured skill handling, simple opponent movement prediction, autonomous robot training and others. Students also presented two graphical user interfaces – one for environment tuning and debugging, other for improved robot training.

Presentations were enhanced by show of robot skills performance. Our students improved some old movement sequences and added few additional combined movements. Some movements were fifty per cent faster than last year movements. New skills included mainly dynamic walking in all directions and work with ball.

Distinctive part of this presentation was contributed by two groups of students, working on massive multi-agent simulation. They revealed a simulation of demonstrating crowd with police interaction. Our knowledge of agent activities had been recast to crowd behaviour model. This model can be reused back in RoboCup simulations.

The extension of the soccer game simulation to the third dimension and other applications shows the continuous progress in RoboCup and in our students' skills, too. Decision making of these robots is very complex and brings new challenge to everyone concerned. We hope that exhibition of robotic and crowd simulation will attract many present and future students and give them motivation for their study and research work.

More information about our annual tournament can be found on the web page `http://www.fiit.stuba.sk/robocup/`.

Programming Contest at IIT.SRC 2013

Peter TREBATICKÝ[*], Mária BIELIKOVÁ[†]

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
{trebaticky,bielik}@fiit.stuba.sk

Abstract. Programming contests has a long tradition at the Slovak university of Technology in Bratislava. As the student research conference offers an open day without any lectures for all our students, we are looking for ways how to attract them. Now, sixth year we have prepared an accompanying event – the programming contest for all our students.

1 Background of the Contest

Programming contests have a long tradition at our university and the faculty. From the beginning in 1998 local contests were organized for our students in order to form teams to represent the Slovak University of Technology in Bratislava at the ACM International Collegiate Programming Contest (ICPC) for Central Europe region. Since 2002 our faculty participates in organization of Czech Technical University Open, which is joint event where universities of Czech and Slovak Republic compete with the aim to select their respective representatives for ACM ICPC Central Europe region.

We prepare our students for this type of programming contest already before they enter the university. We organize for our future students the ProFIIT contest since 2004. It consists of two rounds. In the correspondence round the contestants compete in solving several (around 10) programming problems. They are allowed to compete either on their own or in pairs. The best teams advance into onsite round organized at our faculty. They compete on their own in this round as they can gain bonus points into the admission process. This year is the second time we moved the final round of ProFIIT to coincide with the IIT.SRC in order to show our potential future students exciting research opportunities awaiting them at our faculty. The main reason for this move was that many high school students have only hazy idea what are the projects they will be able to work on during their university study.

Students at our faculty can choose an elective course *Construction of Effective Algorithms* which further develops the algorithmic thinking in them and teaches them the more advanced techniques specifically usable in programming contests. We prepare four 3 hour contests during this one semester course. Participants gain bonus points in them, but these contests are not limited to course participants, everyone can compete for fun. Moreover, our bachelor students selected for the research track have more possibilities in algorithms training, mainly in seminar on advanced algorithms.

[*] Institute of Applied Informatics, Faculty of Informatics and Information Technologies STU in Bratislava
[†] Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

2 Structure of Programming Contest at IIT.SRC

The structure of the programming contest at IIT.SRC closely copies the structure of ACM ICPC. Contestants compete on their own onsite in our computer labs. They have two hours to solve four problems. Problems contain a basic description of what should be solved, exactly specify the format of textual input to the program as well as the format of output and the end of problem statement is the sample input and corresponding sample output.

The task of contestants is to create a program in either C/C++ or Pascal that transforms test input, which has described format but is unknown to contestant, into correct output according to problem statement and in correct format. They submit the source code through our system for programming contests, which compiles the code, runs it against test input, evaluates the given output and informs the contestant of the result. Result is only in the form of simple statement, e.g. "Accepted", "Wrong answer" or "Presentation error" which means the output is not formatted correctly but otherwise appears to have given the correct answer.

The order of contestants is primarily determined by the number of solved problems and in the case of tie, by the sum of the times taken to solve each problem since the beginning of the contest. There is also a 10 minutes penalty for each submitted incorrect solution, but only for the eventually solved problems. This type of order determination favors of course primarily those who solve more problems, but secondarily those who first solve easier problems and also those with lower number of incorrect submissions. The ability to decide fast which problem is the easiest one and to create solution without bugs is also very important apart from the ability to come up with working idea. These skills are mainly trained by practice and learning that is where we help the students through activities mentioned here.

The contest is made more attractive for participants by the fact that during last 45 minutes the preliminary results are not updated. This way, one cannot be sure about her final standing until the awards ceremony. The time interval of not displaying preliminary results was chosen in accordance with conference schedule, because there is another contest ending right before the second poster presentations in which the other conference attendants can tip the winner.

More information about our programming contests can be found on the Web:

- ACM programming contest – `http://www.fiit.stuba.sk/acm/`
- ProFIIT programming contest – `http://profiit.fiit.stuba.sk/`

FIITAPIXEL Exhibition at IIT.SRC 2013

Pavol NÁVRAT, Mária BIELIKOVÁ, Ján LANG*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
{navrat,bielik,lang}@fiit.stuba.sk

Abstract. FIITAPIXEL is an initiative of the Faculty of Informatics and Information Technologies that brings together its members (both students and staff) as well as its potential students and alumni in an effort to create, share and judge pictures. It is organized as an ongoing event, where anyone can contribute pictures to certain categories of photographs. The submitted photographs take part in a contest that is organized annually. Besides best photographs, also best photographers are announced based on their success with their photos. The contest has an expert panel of jurors who give their lists of best photos in each category. In parallel, visitors vote for any photo they like and their votes are counted to result in list of best photos according to popular voting. For the fourth time we organized at the IIT.SRC an exhibition of the best pictures this year contest.

1 FIITAPIXEL as an inspiration

FIITAPIXEL is an initiative of the Faculty of Informatics and Information Technologies to contribute in providing to its members, students and staff alike, an inspiring, creative, stimulating environment to study or to work in. Studying is mostly demanding and hard, and so is working at an institution which faces such a level of competition as is the case in the higher education sector in informatics and information technologies related fields in this region of Europe. From Budapest to Prague, from Vienna to Brno, in a relatively close proximity of Bratislava there several respected institutions with a similar scope of interest. Moreover, in the city itself, there are several other competing institutions.

We try to offer something that may make a little difference. By providing a platform and other forms of support, the Faculty creates an environment that allows expressing its members in a completely different way as it is usual in their professional work. Instead of writing programs or designing chips, they get a chance to express themselves by way of pictures. The language of pictures is intended as a language of artistic expression, even when respecting all the limitations given by the simple fact that these professionals in one (informatics related) field are complete amateurs in another (photography) and similar limitations apply when e.g. elements of journalism are involved.

* Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

2 FIITAPIXEL Organization

FIITAPIXEL started in 2009 and it has been organized ever since then. It takes place as a contest organized annually. The final results are usually announced and prizes awarded around the time of our student research conference IIT.SRC. Immediately after one year of the contest is closed, themes for the next one are published and the contest is open again. The contest is organized in two legs during one year that last approximately half a year each.

There are usually four themes open for each particular leg, but some of them may be adopted for the next period. For example, in 2012/13 contest there were these four themes for the first leg (Summer and Autumn):

- *Enchantment of tininess*
- *Colourful nature*
- *Utterance about a human*
- *The place where I am right now*

with *Images from the street* replacing the third one for the second leg (Winter and Spring).

Each participant can submit up to five pictures to each category both in the first and the second legs. These up to 40 pictures are published on the contest portal, where they are freely visible from anywhere in the world. Anyone can express her/his likes which are treated as votes for the particular picture. At the end of each period, votes are simply counted and the best dozen pictures are announced as winners, according to a popular vote, in each category.

There is also an expert jury formed by experts in visual arts which gives its opinion resulting in another set of lists of dozen winning photos in each category. Results of both opinions, expert and popular, are then used to determine a list of best photographers based on how their photos are placed in particular results.

In the 2012/2013 contest, we have had 1 385 pictures taken by 164 authors. They received nearly 2 700 votes from visitors. Pictures and wining photos are available on the contest portal: `http://foto.fiit.stuba.sk`.

3 IIT.SRC Exhibition

Annual evaluation of the best photographers of the FIITAPIXEL Contest takes place at the student research conference award ceremony. Moreover, we give conference participants the opportunity to enjoy an exhibition of the winning photos of each category in both legs, i.e. we exhibit two dozens of winning pictures, in 2013 in nice new building of the Faculty. IIT.SRC participants can cast their vote for the best photo during the conference. At the end of the day, winner of the participants'vote is announced and awarded.

FIITAPIXEL brings new dimension into our living space at the Faculty together with much inspiration for our activities. The selected best photos will decorate our environment in new building.

High School Students at IIT.SRC Junior 2013

Mária BIELIKOVÁ, Jakub ŠIMKO, Dušan ZELENÍK*

Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies
Ilkovičova, 842 16 Bratislava, Slovakia
{bielik,jsimko,zelenik}@fiit.stuba.sk

Abstract. The IIT.SRC Junior track is a platform for talented high school students interested in informatics and information technologies to present their innovative ideas and projects to their senior colleagues – university students and staff. During poster sessions, works accepted to the IIT.SRC Junior track have been presented by their authors, who subsequently received feedback on their projects throughout discussions.

1 Seeking the talent

Seeking for the talented high school students is essential for maintaining quality of future IIT.SRC conference submissions as well as the life of the faculty. Therefore, we repeatedly started the IIT.SRC Junior track – a platform for high school students to present and discuss their innovative ideas and projects in the field of informatics and information technologies. Last year of IIT.SRC showed to be promising since we managed to involve several talented high school students who recently became our students.

Student works accepted to this track have been presented by their authors during regular poster sessions. Here, the authors had the opportunity to receive valuable feedback from the faculty members as well as from their older colleagues. The authors had also the opportunity to view and discuss other works presented at the conference to gain experience and inspiration for their future projects.

This year, six submissions were selected, what in quantity doubles the pioneer track from last year. Two of these projects are also presented as extended abstracts for more detailed explanation of proposed ideas and realized prototypes. First project authored by Richar Kakaš deals with the aggregation of RSS news and recommendation of articles. Second project authored by Peter Brecska and Mário Kuka faces the problem of measuring and securing by IP cameras and efficiency due to video stream which is transferred to the server. Four other projects are aimed to aggregate items from various bazaars, automate price creation for sellers using exploration of e-shops, manage personal time by analyses of user's week and manage personal finances by recording expenses and income.

More information about the IIT.SRC Junior track can be found on the Web:
http://junior.fiit.stuba.sk/

* Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies STU in Bratislava

IIT.SRC 2013, Bratislava, April 23, 2013, pp. 473–474.

2 IIT.SRC Junior 2013 projects

2.1 Method for personalized aggregation of RSS news

Richard Kakaš, Gymnázium Jura Hronca, Bratislava

Proposed web service merges the benefits of web syndication and recommendation system. This service is downloading articles from RSS/Atom feeds, list of feeds is created and updated by us (of course user can send suggestions). Feeds for many pages and many article categories will be added over time. Therefore, everyone will find his topic on our web site without the need to create own list of RSS/Atom feeds. It also provides filtering by feed domain, time range and article category.

2.2 Automated measuring and control

Peter Brecska, Mário Kuka, SPŠ Franitška Hečku, Levice

Presented work is mostly in increasing the efficiency of existing systems. We modified the safety camera system since it worked inefficiently and put a strain on the entire network (before our modifications, the flow of data over the network was over 3200 kbit/s, after our configurations adjustment and transition to new technology, the flow of data was only 160 kbit/s).

2.3 Aggregation of items offered in various e-bazaars

Slavomír Glinský, Gymnázium Stropkov, Stropkov

Bazarbot.sk is a web application dedicated to aggregate search results from various internet bazaars. It analyzes the given query and crawls to the target bazaars search boxes where the query is submitted. Results sets are then aggregated and displayed at one place. This web application also analyses the query text to extract special name entities such as city or price to fill different search forms at various supported e-bazaars.

2.4 Automated price creation using actual prices in e-shops

Patrik Illith, SŠ Tilgnerova High School, Bratislava

The project is on automated process of price creation. A user is able to sell items using the web application which enables him to automatically adapt the price to compete with other e-shops. We use other internet sites used to aggregate prices from various products such as heureka.sk. We discover the price of the item offered by the seller and keep changing it according to other e-shops automatically using rules made by the seller.

2.5 Web-based support of time management

Marián Pukaj, Gymnázium sv. Andreja, Ružomberok

Presented application enables the user to record all activities he does during a day such as sleeping, working, doing sports. The user also can create rules for automated creation of time spans for activities which are repeating. We display analyses and charts to help the user to understand his time usage. Our intention is to help people to keep time – one of the most valuable things they have.

2.6 Finance management for common users

Miloš Švaňa, Gymnázium Kysucké Nové Mesto, Kysucké Nové Mesto

Project Monee is focused on financial expenses and incomes. It enables the user to track where he spends most of the resources and help to discover problems related to finances. The user continuously records every expense what is used by our method for generating analyses in form of charts. User is then able to discover problems such as inefficient expenses, debts or losses.

Index

PROCEEDINGS IN INFORMATICS AND INFORMATION TECHNOLOGIES

Mária Bieliková (Ed.)

IIT.SRC 2013: Student Research Conference in
Informatics and Information Technologies
Post-Conference Proceedings

1st Edition, Published by
Slovak University of Technology in Bratislava

496 pages, e-print: http://iit-src.stuba.sk
Print Nakladateľstvo STU Bratislava
2013

ISBN 978-80-227-4111-8

ISBN 978-80-227-4111-8