Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií
Študijný program: Softvérové inžinierstvo

Bc. Ladislav Rado

Zdieľanie informácií v portáli založenom na webe so sémantikou

Diplomová práca

Vedúca diplomovej práce:
prof. Ing. Mária Bieliková, PhD.

december 2008

ANOTÁCIA

Slovenská technická univerzita v Bratislave
FAKULTA INFORMATIKY A INFORMAČNÝCH TECHNOLÓGIÍ
Študijný program: Softvérové inžinierstvo

Autor: Bc. Ladislav Rado
Diplomová práca: Zdieľanie informácií v portáli založenom na webe so sémantikou
Vedenie diplomovej práce: prof. Ing. Mária Bieliková, PhD.
december 2008

Súčasná reprezentácia dát na webe predstavuje pre automatické strojové spracovanie problém, ktorý sa web so sémantikou snaží vyriešiť. Hlavným cieľom tejto práce je objavenie komunít výskumníkov so spoločnými záujmami z metadát v doméne publikačnej činnosti. Na reprezentáciu metadát a zápis sémantických vzťahov používame štandardný jazyk OWL. Navrhli sme metódu pre objavovanie komunít založenú na kombinácii dvoch metód: PageRank pre ohodnocovanie autoritatívnosti publikácií a autorov a pravdepodobnostná latentná sémantická analýza pre objavovanie komunít s využitím grafu vzájomných citácií a klasifikácie publikácií. Komunity autorov prezentujeme v portáli, ktorý je založený na webe so sémantikou v textovej a grafickej forme.

Kľúčové slová: web so sémantikou, metadáta, komunity výskumníkov, graf citácií

ANNOTATION

Slovak University of Technology
FACULTY OF INFORMATICS AND INFORMATION TECHNOLOGIES
Degree course: Software Engineering

Author: Bc. Ladislav Rado
Thesis: Sharing Information in a Portal Based on the Semantic Web
Supervisor: prof. Ing. Mária Bieliková, PhD.
2008, December

Current data representation on the web is a problem for machine processing, which the semantic web tries to solve. The main aim of this thesis is to discover communities of researchers with common interests from the metadata in the domain of research publications. We use Web Ontology Language (OWL) for metadata representation and for describing semantic relationships. We have designed a method for discovering communities based on a combination of two methods: PageRank for ranking authoritativeness of publications and authors, and Probabilistic Latent Semantic Analysis for discovering communities using citation graph and classification of the publications. Communities of authors are presented in a portal which is based on semantic web in a textual and graphical form.

Keywords: semantic web, metadata, communities of researchers, citation graph

Prehlásenie

Čestne prehlasujem, že som diplomovú prácu vypracoval samostatne s použitím uvedenej literatúry.

Bratislava, december 2008

Poďakovanie

Ďakujem pani profesorke Márii Bielikovej za vedenie projektu a za cenné pripomienky pri recenzii práce. Ďakujem tiež rodičom za podporu a trpezlivosť.

Obsah

1 Úvod

Web ako sieť, ktorá sa skladá z dokumentov prepojených odkazmi na iné dokumenty (hypertext) umožňuje ľuďom publikovať rôznorodé informácie napríklad pre účely prezentovania nových poznatkov. Problémom súčasného webu je náročnosť automatického strojového spracovania údajov. Publikované dokumenty boli primárne určené pre to, aby ich čítali ľudia. Preto súčasný jazyk používaný na reprezentovanie hypertextových dokumentov (HTML) sa sústreďuje na formátovanie dokumentu. Web so sémantikou je rozšírenie súčasného webu, ktoré umožňuje efektívne spracovanie informácií strojmi s cieľom ich sprístupnenia ľuďom. Namiesto klasického prístupu kedy sú metadáta reprezentované špecializovanými databázami, ontológie poskytujú iný spôsob reprezentácie metadát, taký aby umožňoval univerzálny prístup.

Metadáta sú dáta opisujúce iné dáta. Môžu reprezentovať jednoduché pomenovanie informácií o zdroji, ale aj zložitejšie štruktúrované záznamy. Napríklad 84949 je číselný údaj, ktorého význam bez ďalších súvislostí nie je známy. Pridaním metadát môžeme z čísla 84949 získať poštové smerovacie číslo (PSČ) Bratislavy, to znamená, že sme číslo opísali ako poštové smerovacie číslo.

Hlavným cieľom tejto práce je navrhnúť a overiť metódu na objavenie skupín výskumníkov s podobnými záujmami a umožniť zdieľanie ich výsledkov. Na splnenie tohto cieľa treba ešte:

1. analyzovať štruktúru metadát pre oblasť publikačnej činnosti
2. navrhnúť úpravy metadát, aby bolo možné reprezentovať komunity výskumníkov.
3. prezentovať výsledné skupiny formou portálu založeného na webe so sémantikou.

Táto práca je rozdelená nasledovne: kapitola 2 predstavuje web so sémantikou. Kapitola 3 predstavuje metódy hodnotenia autoritatívnosti publikácií. Kapitola 4 sa venuje vyhľadávaniu komunít. Kapitola 5 skúma známe portálové riešenia v doméne publikačnej činnosti. Kapitola 6 predstavuje návrh metódy hľadania komunít autorov. Kapitola 7 sa venuje získaniu a predspracovaniu metadát. Kapitola 8 analyzuje metadáta objavovaním komunít. Kapitola 9 sa venuje návrhu a implementácii portálu. Kapitola 10 zhodnocuje dosiahnuté výsledky projektu. Okrem toho práca obsahuje dve prílohy: technickú dokumentáciu a médium, ktoré obsahuje produkt s dokumentáciou.

2 Web so sémantikou

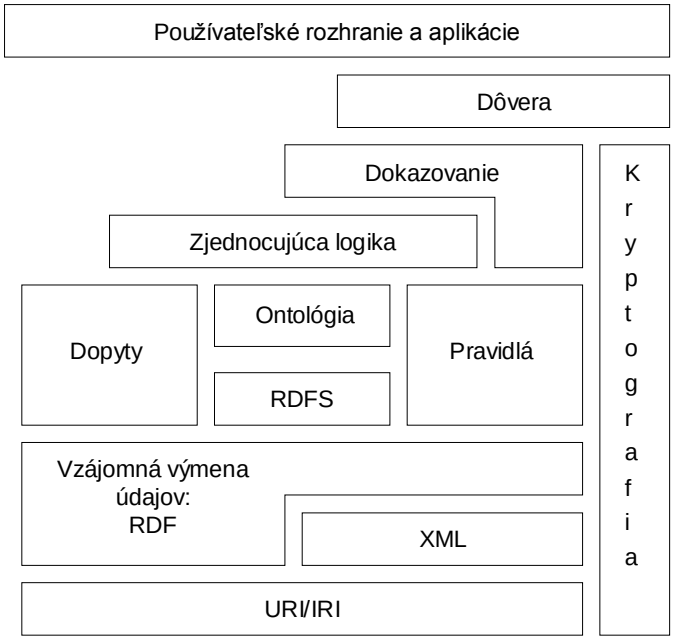
Web so sémantikou je vízia informačného priestoru, ktorý je zrozumiteľný pre stroje, ktoré tak môžu vykonávať úlohy spojené s hľadaním, zdieľaním a kombinovaním informácií na webe. Druhy entít vo svete a vzťahy medzi nimi možno opísať ontológiou.

Pojem ontológia (pôvodne pochádza z filozofie a znamená náuku o bytí) v počítačových alebo informatických vedách znamená formálnu reprezentáciu množiny konceptov v rámci domény a vzťahov medzi týmito konceptmi.

V informatike je často používaná definícia ontológie [Studer *et al.* 1998]: „Ontológia je explicitná formálna špecifikácia zdieľanej konceptualizácie." Konceptualizácia je systém pojmov, ktorý modeluje určitú časť sveta a ten musí byť špecifikovaný explicitne. Formalizácia znamená použitie jazyka so špecifikovanou syntaxou prípadne aj sémantikou. Zdieľaná ontológia nie je individuálnou záležitosťou, ale je výsledkom súhlasu určitej skupiny ľudí.

Ontológia sa používa na uvažovanie o vlastnostiach domény a môže byť použitá na definovanie domény. Je využívaná v oblasti umelej inteligencie, webe so sémantikou, softvérovom inžinierstve a iných vedných disciplínach.

Obrázok 1 znázorňuje vrstvy modelu webu so sémantikou.

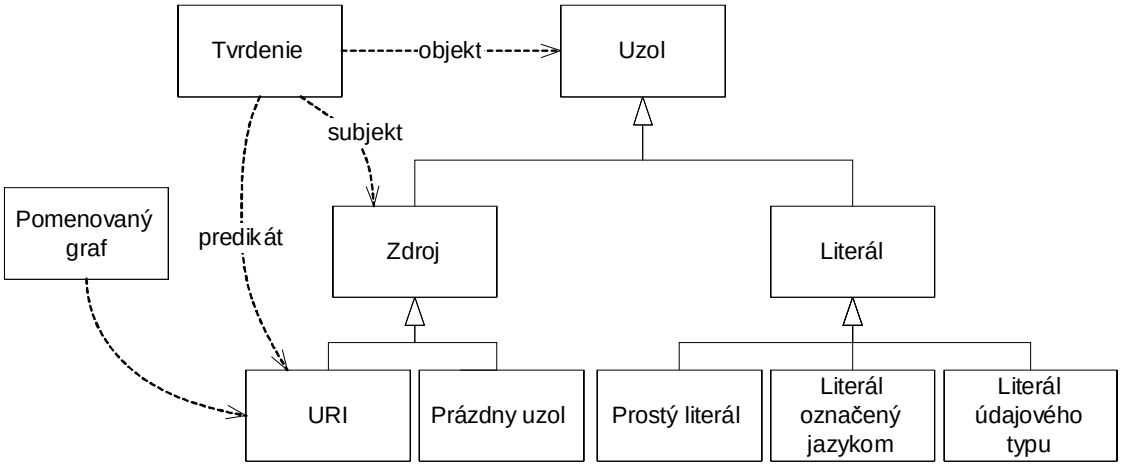


Obrázok 1: Vrstvy modelu webu so sémantikou podľa World Wide Web Consortium

Resource Description Framework (RDF) je jazyk všeobecného účelu pre reprezentáciu informácií na webe. Myšlienkou modelu RDF metadát je posunúť web bližšie k webu so sémantikou a pridať tak do webových stránok štruktúru a pre počítače zrozumiteľný význam (sémantiku). Na vyjadrenie významu a vzťahov sa používa XML zápis trojíc formou výrazu subjekt-predikát-objekt. RDF sa často používa na reprezentovanie osobných informácií, spoločenských sietí, metadát o digitálnych artefaktoch a tiež v zmysle integrácie rôznorodých zdrojov informácií [Breslin *et al.* 2005]. Obrázok 2 znázorňuje RDF model.

Zdrojom (angl. resource) v tomto kontexte môže byť čokoľvek, čo môžeme identifikovať. RDF možno vyjadriť ako graf, ktorý tvoria uzly a orientované hrany. Zdroj má svoj identifikátor v jednotnom tvare nazývaný Unified Resource Identifier (URI). Rozšírenie International Resource Identifier (IRI) pridáva možnosť zapísať identifikátor znakmi z Unicode. Zdroje majú vlastnosti (angl. property). V diagrame ich označujeme hranou s názvom (angl. label), ktorá je orientovaná v smere od zdroja k hodnote vlastnosti. Ak je vlastnosť označená ako funkcionálna, môže nadobúdať len jednu hodnotu ako matematická funkcia. Vlastnosť okrem toho môže byť buď objektová (vtedy je zdroj ako identifikátor), alebo údajového typu (literál).

Každá vlastnosť má hodnotu, ktorá môže byť buď reťazec znakov, označený ako prostý literál (angl. plain literal) alebo reťazec znakov s označením jazyka, v ktorom je zapísaný (angl. language tagged literal), napríklad "číslo"@"sk", "number"@"en", alebo reťazcom zapísaný údajový typ (datatyped literal) napríklad číslo "12.345"^^xsd:float.



Obrázok 2: Web so sématikou: RDF model

Resource Description Framework Schema (RDFS), ako jazyk pre opis slovníka (vocabulary description language), je rozšírením RDF. Navyše poskytuje možnosť odvodzovania definičného oboru (domain) a oboru hodnôt (range). Zavádza včlenenie (subsumption) pre vlastnosti a triedy.

Jedným zo štandardných jazykov pre reprezentáciu ontológií je Web Ontology Language (akronym OWL), ktorý navyše od RDFS podporuje pridanie ohraničení na existenciu definičného oboru a/alebo oboru hodnôt. Ontológia zapísaná v *OWL* môže obsahovať opisy *tried*, *vlastností* a ich *inštancií*. Sémantika OWL špecifikuje, ako odvodiť dôsledky, t.j. fakty, ktoré nie sú prítomné v ontológii, ale sú spôsobené sémantikou.

RIF (Rule interchange format) je formát pre výmenu pravidiel v systémoch založených na pravidlách pre web so sémantikou. Cieľom je vytvoriť formát pre rôzne jazyky s pravidlami, ktoré sa použijú pri odvodzovaní znalostí.

SPARQL je jazyk pre dopyty pre RDF, ktorý pokrýva požiadavky identifikované v prípadoch použitia[1], pre ktoré bol tento jazyk navrhnutý.

Ontológie predstavujú iný spôsob reprezentácie dát, ako napríklad databázy, ktoré sú špeciálne vytvorené pre určité údaje.

Použitím štandardného formátu pre reprezentovanie metadát a použitím ontológií chceme docieliť, aby stroje boli schopné spracovať metadáta.

[1] Prípady použitia sú na stránke http://www.w3.org/TR/rdf-dawg-uc/

3 Hodnotenie autoritatívnosti publikácií

V tejto časti uvádzame niekoľko známych metód pre hodnotenie autoritatívnosti. Tieto metódy hodnotia uzly grafu, čo možno použiť na hodnotenie publikácií a autorov, reprezentovaných v grafe vzťahmi ako *spoluautor*, *cituje* a inými.

3.1 PageRank

Pomocou metódy PageRank je možné vypočítať relatívne hodnotenie pre každý uzol grafu. Možno tým hodnotiť napríklad webové stránky, čo má využitie vo vyhľadávaní. Autori [Page *et al.* 1998] opísali metódu hodnotenia uzlov ako šírenie hodnotenia cez hrany: „Uzol má vysoké hodnotenie, ak súčet hodnotení od naň smerujúcich uzlov je vysoký." Vzaté na publikácie, tvrdenie platí jednak pre prípad, ak publikáciu cituje mnoho publikácií a tiež pre prípad, ak publikáciu cituje málo publikácií s vysokým hodnotením.

Formálne nech $G(V, E)$ je graf daný množinou uzlov V a množinou orientovaných hrán E. Vybraný uzol grafu nech je v. Odkazy na uzol môžu pribúdať s vývojom siete v čase, vo všeobecnosti možno o uzle s určitosťou povedať len aké má všetky priame (odchádzajúce) odkazy, ale nie aké sú všetky spätné (prichádzajúce) odkazy. Počas výpočtu hodnotenia sa graf nemení pridávaním alebo odoberaním uzlov a tiež sa nemenia hrany medzi uzlami. Hodnotenie PageRank je vypočítané podľa vzorca:

$$r_v^{i+1} = \frac{d}{n} + (1 - d) \sum_{u:(u,v)\in E} \frac{r_u^i}{\sigma_u}$$

kde σ_u je počet hrán vychádzajúcich z vrcholu u, d je damping faktor, n je počet všetkých vrcholov grafu.

Výpočet hodnotenia možno vykonať iteratívne voľbou vektora s ľubovoľnými počiatočnými hodnoteniami. Výsledný vektor je aproximáciou hodnotenia PageRank pre každý uzol. Voľba štartovacieho vektora nemá vplyv na výsledné hodnotenia, ale na rýchlosť konvergencie. Dobrou voľbou štartovacieho vektora bude pre dosiahnutie požadovanej presnosti potrebných menej iterácií. Časová zložitosť výpočtu je priamo úmerná počtu hrán a iterácií $\mathcal{O}(|E| * i)$. Pre urýchlenie konvergencie možno použiť iné numerické metódy. Z hľadiska vlastnosti konvergencie je výhodou algoritmu dobrá škálovateľnosť pre rozsiahle grafy.

PageRank patrí k spektrálnym metódam [Flake *et al.* 2003]. Nech A je matica susednosti, ktorej prvky vyjadrujú existenciu orientovanej hrany z vrcholu i do j kedy platí $A_{i,j}$ inak 0. Matica M nech je normalizovaná matica A tak, že súčty v riadkoch dávajú jedna. Nech U je matica rozmeru $n \times n$, ktorej všetky prvky sú rovné jedna. Vektor PageRank je potom rovný maximálnemu vlastnému vektoru matice

$$(dU + (1-d)M)^T$$

ktorý možno hľadať napríklad mocninovou metódou.

3.2 PageRank s prednostnými uzlami

Rozšírenie PageRank umožňuje uprednostniť niekoľko uzlov [White & Smyth 2003]. Množina počiatočných uzlov je zvolená vopred. S určitou pravdepodobnosťou sa simulovaný náhodný pochod grafom preruší a začína sa výberom uzla z množiny počiatočných uzlov, ktoré sú takto uprednostnené. Výpočtová zložitosť je rovnaká ako pri pôvodnom algoritme PageRank.

$$r_v^{i+1} = dp_v + (1-d) \sum_{u:(u,v)\in E} \frac{r_u^i}{\sigma_u}$$

kde vektor počiatočných preferovaných uzlov s hodnotami $p_v = \frac{1}{|R|}$ pre $v \in R \subseteq V$ a $p_v = 0$ inak.

3.3 HITS

HITS (Hyperlink-Induced Topic Search) [Chakrabarti *et al.* 1999, Kleinberg 1999] algoritmus počíta hodnotenie *autorita* a *rozbočovač* pre každý uzol grafu. Algoritmus pozostáva z dvoch hlavných krokov. Prvým krokom je výber niekoľkých stránok, ktoré sú zrejme relevantné autority. Druhým krokom je šírenie váhy, ktoré rozhoduje o rozbočovači a autorite iteratívnym procesom.

Opísané podrobnejšie, HITS začína konštrukciou časti grafu (podgrafu) na základe dopytu do vyhľadávača. O vrátenej kolekcii stránok vyhľadávačom sa predpokladá, že aspoň niektoré stránky budú relevantné tým, že budú obsahovať odkazy na autority. Podgraf nemusí nevyhnutne obsahovať autoritatívne stránky. Potom sa sieť expanduje na základe odkazov zo stránok v pôvodnej množine. Ďalej sa takto pracuje s podgrafom stránok s tým, že odkazy v rámci rovnakej domény sú považované za navigačné a neprispievajú k hodnoteniu autority. Výsledkom je priradená nezáporná hodnota *autorita* a *rozbočovač* každému uzlu. Keďže nás zaujíma

relatívne usporiadanie podľa váhy, aplikuje sa normalizácia, tak že celkový súčet je ohraničený.

$$a = A^T h$$
$$h = Aa$$
$$a \leftarrow a/\|a\|_2$$
$$h \leftarrow h/\|h\|_2$$

kde a je vektor autorít a h je vektor rozbočovačov, A je matica susednosti citácií.
Lepšia *autorita* má vyššiu váhu a, a podobne lepší *rozbočovač* má vyššiu váhu h. Autori nastavili počiatočné hodnotenie na rovnaké hodnoty, pretože nerobili vopred odhad hodnotení.

3.4 Kombinácia HITS a PageRank

Metódu HITS a PageRank možno kombinovať rozšírením metódy HITS o model náhodného surfera z metódy PageRank [Nie *et al.* 2007]. Hodnotenie rozbočovač, resp. autorita sa potom distribuuje medzi potomkov, resp. rodičov uzla. Okrem toho je do modelu zahrnutý aj náhodný skok na ľubovoľný iný uzol grafu s určitou pravdepodobnosťou. V takomto rozšírení môže náhodný surfer chodiť po doprednych aj spätných odkazoch.

Pravdepodobnosť stránky v, že je navštívená smerom po doprednom odkaze je vo vzorci jej autorita $A(v)$ a pravdepodobnosť, že stránka je navštívená po spätnom odkaze je jej rozbočovač $H(v)$.

$$A(v) = \frac{d}{n} + (1-d) \sum_{u:(u,v)\in E} \frac{H(u)}{\sigma_u}$$
$$H(u) = \frac{d}{n} + (1-d) \sum_{v:(u,v)\in E} \frac{A(v)}{\iota_v}$$

kde σ_u je počet odchádzajúcich hrán a ι_u počet prichádzajúcich hrán uzla u.
Náhodný skok na ľubovoľný uzol s rovnomerným rozdelením možno nahradiť vektorom preferovaných uzlov. Dostávame tým rozšírenie metódy HITS s prednostnými uzlami.

3.5 CiteRank

Metóda hodnotenia publikácií *CiteRank* [Walker *et al.* 2007] vychádza z metódy hodnotenia webových stránok PageRank. CiteRank modeluje proces hľadania publikácií výskumníkom. Na rozdiel od metódy PageRank, kde autor začína na náhodne vybratej

stránke, v sieti publikácií výskumník začína vyhľadávanie od súčasných publikácií napríklad sledovaním aktualizácie archívu publikácií alebo nedávno vydaným žurnálom. Preto realistickejší model by mal brať do úvahy to, že vedci preferujú začiatok hľadania od súčasných publikácií a pokračujú postupne smerom k čoraz starším publikáciám.

CiteRank je dvojparametrový model, ktorý umožňuje odhadnúť premávku $T_i(\tau, \alpha)$ pre danú publikáciu i. Z celej množiny publikácií sa náhodne vyberie publikácia s pravdepodobnosťou exponenciálne znižovanou podľa veku publikácie s charakteristickým časom útlmu τ. V každom kroku na ceste hľadania publikácií je s pravdepodobnosťou α vedec spokojný a zastaví cestu prieskumu. S pravdepodobnosťou $1 - \alpha$ je nasledovaná náhodne vybratá citácia susednej publikácie.

CiteRank možno interpretovať ako relevanciu publikácie v kontexte spôsobu vyhľadávania publikácií. Na druhej strane PageRank je o hodnotení ako o ocenení celoživotnej autorovej práce. Porovnanie CiteRank s tradičnou metódou hodnotenia publikácií ako dosiahnutý počet citácií ukazuje, že tieto dve metódy majú vysokú mieru podobnosti – čím vyšší počet citácií má publikácia, tým pravdepodobnejšie ju navštívi výskumník jedným z prichádzajúcich odkazov.

V CiteRank citácia z vysoko hodnotenej publikácie prispieva k hodnoteniu citovanej publikácie. Vek citujúcej publikácie je braný do úvahy tak, že účinok nedávnej citácie na publikáciu je väčší, ako staršej citácie na rovnakú publikáciu. Nové citácie indikujú relevanciu publikácie v kontexte súčasných smerov výskumu.

Formálne možno model opísať nasledovne: Nech W je matica asociovaná so sieťou citácií. Prvky matice $W_{i,j} = \frac{1}{\sigma_j}$ ak j cituje i a 0 inak, kde σ_j je výstupný stupeň publikácie j. Nech ρ_i je pravdepodobnosť výberu publikácie i ako prvej daná vzťahom $\rho_i = e^{-t_i/\tau}$, kde t je vek publikácie (napríklad v rokoch). Pravdepodobnosť, že výskumník vyberie ako prvú publikáciu pri výbere jedinej publikácie je daná vektorom $\vec{\rho}$. Podobne pravdepodobnosť nájdenia publikácie po nasledovaní jedného odkazu je $(1 - \alpha)W \cdot \vec{\rho}$. Celková premávka publikácie je definovaná ako pravdepodobnosť narazenia na ňu na ceste ľubovoľnej dĺžky:

$$\vec{T} = I \cdot \vec{\rho} + (1 - \alpha)W \cdot \vec{\rho} + (1 - \alpha)^2 W^2 \cdot \vec{\rho} + \cdots$$

Prakticky možno vypočítať CiteRank zobratím postupných členov rozvoja do dostatočnej konvergencie (napr. $< 10^{-6}$ priemernej hodnoty).

3.6 Zhodnotenie

Uvedené metódy predstavujú hodnotenie autoritatívnosti publikácií a vychádzajú z grafu citácií. Kým metóda PageRank hodnotí súhrnnú prácu autora, metóda CiteRank dáva do popredia publikácie, ktoré sú relevantné v kontexte súčasných smerov výskumu.

Metóda HITS vyhľadáva okrem *autorít* aj uzly, ktoré odkazujú na autority nazývané *rozbočovače*.

PageRank aj HITS môžu byť použité na detekciu komunít [Flake et al. 2003]. V prípade metódy PageRank s prednostnými uzlami za komunitu považovať všetky uzly, ktoré majú hodnotenie vyššie ako stanovená hraničná hodnota. V prípade metódy HITS možno analyzovať nemaximálne vlastné vektory matíc AA^T a A^TA.

Metódy relatívneho hodnotenia uzlov HITS a PageRank a obdoba CiteRank (nazývaná K-krokový Markovovský proces) boli implementované v knižnici JUNG[2]. Okrem toho sa v tejto knižnici nachádza metóda pre výpočet vzdialenosti medzi dvoma vrcholmi pomocou Dijkstrovho algoritmu.

[2] Java Universal Network Graph http://jung.sourceforge.net/

4 Vyhľadávanie komunít

Hlavným cieľom práce je objavovanie skupín výskumníkov s podobnými záujmami, ktoré vlastne predstavujú určité komunity.

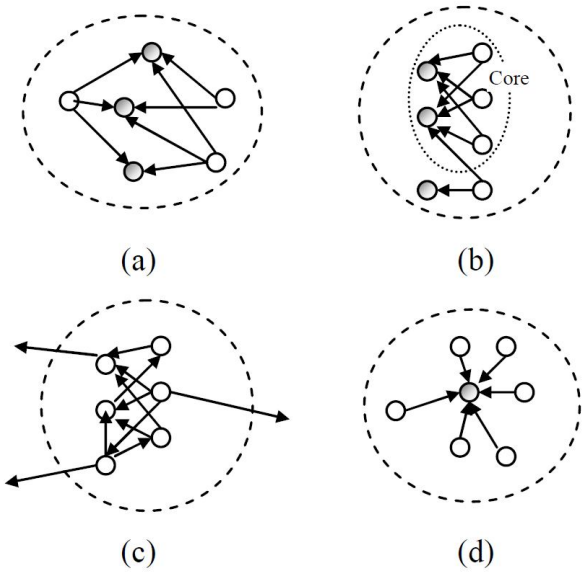
4.1 Definície komunity

Uvádzame tri pohľady na definíciu komunity [Yolum & Singh 2003]:

Sociológia. Pôvodná definícia komunity pochádza z analýzy spoločenských sietí v sociológii. Tradične je komunita definovaná ako skupina ovplyvňujúcich sa ľudí, žijúcich v spoločnej oblasti. Hlavnou úlohou je porozumieť rôznym druhom spoločenských vzťahov medzi ľuďmi.

Statická analýza odkazov. Pre dolovanie komunít z webových stránok bolo nedávno vytvorených niekoľko prístupov Tieto prístupy sa na populácie pozerajú ako na grafy, kde hrany nemajú značky a v rámci modelu sa nemenia. Komunity sú definované ako vzory vzájomnej podobnosti.

Referencie a prispôsobivosť. Tieto prístupy uvažujú interakcie medzi agentmi (alebo ľuďmi, ktorých môžu reprezentovať). Agenty (stroje) udržiavajú svoj model a pomáhajú nájsť iné agenty podaním referencií. Agenty sa potenciálne učia o každom a adaptívne rozhodujú, ktorých z iných agentov budú mať za susedov.



Obrázok 3: Rôzne pohľady na komunity [Zhou *et al.* 2002]

Obrázok 3 (na predchádzajúcej strane) znázorňuje rôzne pohľady na komunity [Zhou *et al.* 2002] (a) bipartitný graf s niekoľkými autoritatívnymi stránkami navzájom prepojenými, ktoré zdieľajú spoločnú tému, (b) jadro s úplným bipartitným grafom (c) viac prepojení v rámci komunity ako mimo nej, (d) okolie citovanej publikácie.

Ako sa uvádza v literatúre [Lausen *et al.* 2005] medzi skupinami a komunitami existujú určité rozdiely. Skupiny majú zvyčajne definovanú vnútornú štruktúru a administratívne pravidlá. Ďalšie rozdiely sú uvedené v tabuľke 1:

	Skupina	**Komunita**
Veľkosť	Malá	Rozsiahla
Stupeň interakcie	Viazaný	Voľný
Motivácia	Spoločný cieľ	Spoločné záujmy
Ciele práce	Definované a zdieľané výsledné ciele	Občasná výmena informácií
Personálny vzťah	Osoby sa navzájom osobne poznajú.	Osoby sa nepoznajú

Tabuľka 1: Porovnanie skupín a komunít

4.2 Problém detekcie komunít

Prieskumom a porozumeniu spoločenským vzťahom medzi jedincami sa zaoberá analýza sociálnych sietí (Social Network Analysis) ďalej len SNA. Sociálna sieť je modelovaná grafom, kde uzly reprezentujú jedincov a hrany priame vzťahy medzi nimi.

Zaujímavou oblasťou v rámci SNA je dolovanie komunít (Community Mining). Komunita môže byť definovaná ako skupina entít, ktoré zdieľajú podobné vlastnosti alebo sa navzájom spájajú cez určité vzťahy. Typickými problémom SNA je objavovanie skupín, ktoré zdieľajú podobné vlastnosti a vyhodnocovanie dôležitosti jedincov. V typickej spoločenskej sieti sa vždy nájdu rozličné vzťahy medzi jedincami, napríklad kamarátstvo, obchodné vzťahy a spoločné záujmy.

Vo všeobecnosti sa na analýzu spoločenských sietí aj dolovanie komunít možno pozerať ako na dolovanie grafu. Dolovanie komunity možno považovať za identifikovanie časti grafu. Skoro všetky techniky dolovania grafu a dolovania komunít sú založené na homogénnom grafe, to znamená len jednom type vzťahu medzi objektmi.

Na druhú stranu v reálnych spoločenských sieťach sú medzi objektmi vždy rôzne druhy vzťahov. Takýto druh spoločenskej siete možno nazvať *multi-relačná spoločenská sieť* alebo *heterogénna spoločenská sieť* [Ren *et al.* 2007]. V SNA blízkosť dvoch súvisiacich konceptov v sieti je zvyčajne meraná ako skóre relevancie, ktoré je založené na vybraných vzťahoch medzi entitami.

Na identifikovanie komunít sa možno pozerať ako na hľadanie zhlukov grafu. Použitie klasických metód, ako napríklad rozdeľovanie grafu na vyhľadanie optimálnych zhlukov komunít, predstavuje NP (Non-Polynomial) ťažký problém. Preto všetky navrhované metódy závisia od niektorých vlastností grafov. Niektoré metódy vyhodnocujú len okolie uzlov na to, aby rozhodli či patrí do rovnakej komunity. Iné metódy prehľadávajú celý graf na to, aby našli príslušnosť ku komunite. Bez ohľadu na použitú metódu, je tu rozpor medzi použitými výpočtovými zdrojmi a kvalitou výslednej komunity.

Skoršie prístupy pre identifikovanie komunít môžu byť rozdelené do dvoch kategórií: hierarchický a rozdeľujúci prístup. Hierarchický prístup pracuje tak, že zoskupuje dva najbližšie uzly do jednej komunity, pokiaľ sa celá sieť nestane jedinou komunitou. Rozdeľujúci prístup pracuje zhora nadol, kde rozdeľuje celú sieť na dve komunity, až pokiaľ každý uzol nie je komunitou. Tieto algoritmy zvyčajne potrebujú určitú metriku, aby vyhodnotili blízkosť a rozdielnosť medzi dvoma uzlami. Najpoužívanejší rozdeľujúci prístup je *GN (Girvan-Newman) algoritmus* [Newman & Girvan 2004], ktorý používa metriku *betweenness*. Newman navrhol mieru modularity Q pre vyhodnocovanie toho, aká dobrá je štruktúra komunity. Vysoká hodnota Q zvyčajne indikuje dobrú štruktúru komunity.

4.3 Pravdepodobnostný model

Štatistická technika *pravdepodobnostná latentná sémantická analýza* (PLSA) [Hofmann 2001] umožňuje analýzu vzájomne sa vyskytujúcich dvojíc dát, v tomto prípade autorov a_i, a tém t_j. Model so skrytými premennými zavádza predpoklad, že každé prvky v každej z dvoch množín autorov A a tém T sú nezávisle podmienené od stavu asociovanej skrytej premennej z množiny Z. Pretože počet skrytých premenných je menší ako počet autorov a tém, dochádza k zhusteniu.

Združená pravdepodobnosť modelu je definovaná ako

$$P(a_i, t_j) = P(a_i)P(t_j|a_i), \;\; P(t_j|a_i) = \sum_k P(t_j|z_k)P(z_k|a_i)$$

Pretože parametre modelu nemožno vypočítať priamo, štandardným postupom pre určenie maximálnej vierohodnosti L v modeloch so skrytými premennými je iteratívny Expectation Maximization (EM) algoritmus.

$$L = \sum_i \sum_j N_{i,j} \log P(a_i, t_j)$$

EM algoritmus sa strieda v dvoch krokoch:

1. v E kroku sa vypočíta očakávaná pravdepodobnosť na základe dát pre skryté premenné:

$$P(z_k|a_i, t_j) = \frac{P(t_j|z_k)P(z_k|a_i)}{\sum_{k'} P(t_j|z_{k'})P(z_{k'}|a_i)}$$

2. v M kroku sú aktualizované parametre modelu:

$$P(t_j|z_k) \propto \frac{\sum_i N_{i,j} P(z_k|a_i, t_j)}{\sum_i \sum_{j'} N_{i,j'} P(z_k|a_i, t_{j'})}$$

$$P(z_k|a_i) \propto \frac{\sum_j N_{i,j} P(z_k|a_i, t_j)}{\sum_{j'} N_{i,j'}}$$

Parametre modelu sú inicializované náhodne. Po niekoľkých iteráciách sa na základe stabilizovanej hodnoty logaritmu vierohodnosti L modelu dosiahne konvergencia k lokálnemu maximu. Skupina g pre autora a_i je $g(a_i) = \arg\max_{z_k} P(z_k|a_i)$.

4.4 Pravdepodobnostný model s uvažovaním odkazov medzi publikáciami

PHITS vykonáva pravdepodobnostnú faktorizáciu citácií dokumentov pre bibliometrickú analýzu [Cohn & Hofmann 2001]. Bibliometria sa snaží identifikovať témy v kolekcii dokumentov, a tiež vplyvných autorov a publikácie na tieto témy, založené na vzoroch frekvencie citácií. Táto analýza bola aplikovaná na referencie pre vytlačenú literatúru, ale rovnaké techniky sa ukázali byť úspešné aj pre analýzu hypertextovej štruktúry na webe.

Tradičná bibliometria analyzuje maticu dokument-citácia. Prvok tejto matice označený ako $A_{i,j}$ je nenulový práve vtedy, keď dokument i je citovaný dokumentom j. Koeficient dokumentu v hlavnom vlastnom vektore matice $A^T A$ je interpretovaný ako autorita tohto dokumentu v rámci komunity – ako pravdepodobne bude citovaný v rámci tej komunity. Koeficient dokumentu v hlavnom vlastnom vektore matice AA^T je interpretovaný ako rozbočovač v komunite – koľko autoritatívnych dokumentov ho cituje v rámci komunity.

Pri PHITS pravdepodobnostný model nahrádza analýzu vlastných vektorov štatistickou interpretáciou. PHITS je matematicky identický s PLSA s jedným rozdielom: namiesto modelovania citácií obsiahnutých v rámci dokumentu (čo zodpovedá modelovaniu slov v dokumente pri PLSA), PHITS modeluje vstupné odkazy, citácie smerujúce na dokument. Nahrádza odhadom pravdepodobnosti zdroja citácie $P(c|z)$ odhad pravdepodobnosti slova v PLSA. Pre danú tému určenú faktorom z, pravdepodobnosť, že dokument je citovaný $P(d|z)$ sa dá interpretovať ako autorita dokumentu vzhľadom na túto tému.

Odvodené rovnice pre E krok v EM algoritme:

$$P(z_k|t_i, d_j) = \frac{P(t_i|z_k)P(z_k|d_j)}{\sum_{k'} P(t_i|z_{k'})P(z_{k'}|d_j)}, \quad P(z_k|c_l, d_j) = \frac{P(c_l|z_k)P(z_k|d_j)}{\sum_{k'} P(c_l|z_{k'})P(z_{k'}|d_j)}$$

Pre M krok vypočítané podmienené rozdelenia a zmiešané proporcie

$$P(t_i|z_k) = \sum_j \frac{N_{i,j}}{\sum_{i'} N_{i',j}} P(z_k|t_i, d_j), \quad P(c_l|z_k) = \sum_j \frac{A_{l,j}}{\sum_{l'} A_{l',j}} P(z_k|c_l, d_j)$$

$$P(z_k|d_j) \propto \alpha \sum_j \frac{N_{i,j}}{\sum_{i'} N_{i',j}} P(z_k|t_i, d_j) + (1-\alpha) \sum_l \frac{A_{l,j}}{\sum_{l'} A_{l',j}} P(z_k|c_l, d_j)$$

Logaritmus vierohodnosti tohto modelu je:

$$L = \sum_j \left(\alpha \sum_i \frac{N_{i,j}}{\sum_{i'} N_{i',j}} \log \sum_k P(t_i|z_k)P(z_k|a_j) + (1-\alpha) \sum_l \frac{A_{l,j}}{\sum_{l'} A_{l',j}} \log \sum_k P(c_l|z_k)P(z_k|d_j) \right)$$

4.5 Pravdepodobnostný model pre homogénny graf

Predpokladajme, že graf má neorientované hrany a hrany nemajú váhy. Nech má graf n uzlov a A je matica susednosti. Existencia hrany je modelovaná skrytou informáciou o tom, že uzly, ktoré spája môžu patriť do rovnakej komunity. Nech má byť objavených c komunít, a nech π_r je časť uzlov v komunite r. Komunita r priradí jej váhu $\beta_{r,i}$ uzlu i tak, že $\sum_{i=1}^{N} \beta_{r,i} = 1$. Na $\beta_{r,i}$ sa možno pozerať ako na dôležitosť uzla i v komunite r. Pravdepodobnosť hrany medzi uzlami i a j pod podmienkou, že oba uzly patria do komunity r je modelovaná ako:

$$P(e_{i,j}|r,\pi,\theta) = \beta_{r,i}\beta_{r,j}$$

Táto pravdepodobnosť môže byť považovaná za príspevok komunity r k vytvoreniu hrany $e_{i,j}$. Potom pravdepodobnosť vytvorenia hrany $P(e_{i,j}|\pi,\theta)$ je súčet cez všetky komunity $r = 1,2,\dots,c$.

$$P(e_{i,j}|\pi,\theta) = \sum_{r} \pi_r P(e_{i,j}|r,\pi,\theta) = \sum_{r} \pi_r \beta_{r,i}\, \beta_{r,j}$$

Pozorovaná informácia je hrana $e_{i,j}$ avšak bola určená nepozorovanými parametrami $\pi_r, \beta_{r,i}$. Tieto skryté parametre sú to, čo rozhoduje o topológii grafu A. Mali by sme preto odhadnúť tieto nepozorované parametre, aby s najvyššou pravdepodobnosťou vygenerovali graf A. Logaritmus pravdepodobnosti celého grafu môže byť modelovaný ako:

$$L = \log P(A|\pi,\beta) = \sum_{i=1}^{n} \sum_{j:A_{i,j}=1} \log P(e_{i,j}|\pi,\beta)$$

Parametre je zložité odhadnúť pretože rovnica obsahuje logaritmus sumy, ale úloha môže byť optimalizovaná použitím Expectation Maximization (EM) algoritmu. Úplný postup odvodenia rovníc sa nachádza v literatúre [Ren et al. 2007], výsledné obidva kroky algoritmu sú:

Rovnica pre E krok:

$$q_{ij,r} = P(e_{ij\in r}|A,\pi,\beta) = \frac{\pi_r \beta_{r,i}\beta_{r,j}}{\sum_{s=1}^{c} \pi_s \beta_{s,i}\beta_{s,j}}$$

Pravdepodobnosť $q_{ij,r}$ označuje znalosť toho, keby sme vedeli že hrana $e_{i,j}$, patrí do komunity r pri topológii grafu A.

Rovnice pre M krok:

$$\pi_r = \frac{\sum_i \sum_{j:A_{i,j}=1} q_{ij,r}}{\sum_i \sum_{j:A_{i,j}=1} \sum_s q_{ij,s}} \qquad \beta_{r,i} = \frac{\sum_{j:A_{i,j}=1} q_{ij,r}}{\sum_{k=1}^{N} \sum_{j:A_{i,j}=1} \sum_s q_{ij,r}}$$

4.6 *Ďalšie metódy hľadania komunít*

Komunity možno hľadať aj z nemaximálnych vlastných vektorov matíc (Laplaceián) [Capocci *et al.* 2005] avšak len ak sú komunity prepojené viacerými vzťahmi v rámci komunity ako mimo komunity a komunity sa neprekrývajú. Autori túto metódu použili na hľadanie synoným k určitým slovám.

Ďalším prístupom je hľadanie metódou sústredných kružníc [Zhou *et al.* 2002], pri ktorej sa najprv vyhľadajú autority. Hľadanie komunity postupuje expanziou smerom od objektov jadra. Objekt jadra je hľadaný ako taký, na ktorý iné objekty odkazujú niekoľkokrát a pri prekročení určitého hraničného počtu sa objekt považuje za jadro. Objekty vzdialenejšie súvisia s objektmi v jadre, ale so zväčšujúcou vzdialenosťou od nich majú čoraz nižšiu relevanciu.

5 Existujúce portály pre prácu s publikáciami

V tejto kapitole uvedieme niekoľko portálov, ktoré umožňujú vyhľadávanie autorov a publikácií a porovnáme ich najmä z hľadiska možností vytvárania komunít.

- ArnetMiner http://www.arnetminer.org/
- Bibsonomy http://bibsonomy.org/
- DBconnect (University of Alberta) http://kingman.cs.ualberta.ca/research/demos/content/dbconnect/
- Libra (Microsoft Asia) http://libra.msra.cn/
- OpenAcademia http://openacademia.org/
- CiteSeer http://citeseer.ist.psu.edu/
- Rozšírenie ACM portálu http://portal.acm.org/

5.1 ArnetMiner

Systém umožňuje podľa hlavných cieľov autorov nasledovné [Tang *et al.* 2007]:

- Extrakciu profilov autorov z webu na základe webových stránok o autoroch
- Integráciu profilov autorov a ich publikácií
- Vyhľadávanie
    a. Vyhľadávanie profilu autora na základe mena autora. Pre spresnenie hľadaného mena možno ako vstup zadať príslušnosť autora k organizácii.
    b. Vyhľadávanie publikácií: na základe kľúčových slov sa zobrazia relevantné publikácie spolu s odkazom na elektronickú verziu dokumentu.
    c. Vyhľadávanie konferencií podľa názvu konferencie, zobrazenie detailov o konferencii.
- Objavovanie znalostí
    a. Hľadanie expertov na určitú tému. Hodnotenie experta na základe toho, ako často sa jeho meno vyskytuje v autorstve na zadanú tému. Potom nasleduje šírenie hodnotenia na ostatných autorov.
    b. Hľadanie asociácií medzi autormi. Pre hľadanie najkratších spojení medzi dvoma autormi je použitý Dijkstrov algoritmus, ktorý je pre výpočtovú zložitosť zhora ohraničený na vyhľadanie najväčšej dĺžky spojenia.

c. Hľadanie tém – najčastejšie vyskytujúce sa sekvencie slov určujú popularitu témy.
d. Hľadanie prehľadových publikácií pre zadanú tému.

Z implementačného hľadiska sú metadáta reprezentované ontológiou. Ako zdroj dát je použitá množina DBLP[3] (Digital Bibliography & Library Project). Systém neumožňuje vyhľadávanie skupín.

5.2 Bibsonomy

Predstavuje systém pre zdieľanie obľúbených položiek a publikácií. Umožňuje vyhľadávanie na základe značiek, používateľa, skupiny. Zobrazuje súvisiace značky a podporuje export citácií do formátov RSS, BibTeX, RDF.

Umožňuje export do ontologického formátu RDF, používa ontológiu SWRC (Semantic Web Research Communities). Zdrojom dát sú metadáta od používateľov.

Systém umožňuje len manuálne vytváranie skupín.

5.3 DBConnect

Predstavuje systém pre odporúčanie autorov, konferencií a tém na základe metadát publikácií [Zaiane *et al.* 2007]. Zobrazenie metadát je v textovej forme v niekoľkých tabuľkách.

Obrázok 4 znázorňuje podrobné informácie o vybranom autorovi publikácií. Sú tu uvedené počty citácií, najčastejší spoluautori autora, súvisiace konferencie, súvisiaci autori, súvisiace témy, odporúčaní autori. Pri odporúčaných autoroch možno zobraziť najkratšiu asociáciu medzi dvoma autormi a tiež súvisiace témy a konferencie odporúčaného autora. Komunita je tu definovaná ako skupina uzlov navzájom prepojená viacerými spojeniami v rámci komunity ako mimo komunity. Portál používa relačnú databázu, ktorá je naplnená metadátami z DBLP. Portál umožňuje zobrazenie odporúčaní pre autorov, témy a konferencie. Témy sú získané ako často vyskytujúce sa dvojice a trojice slov v nadpisoch publikácií.

Systém umožňuje odporúčanie nových spolupracovníkov vybraného autora, zobrazenie podobných tém a konferencií.

Chýba zobrazenie skupín autorov.

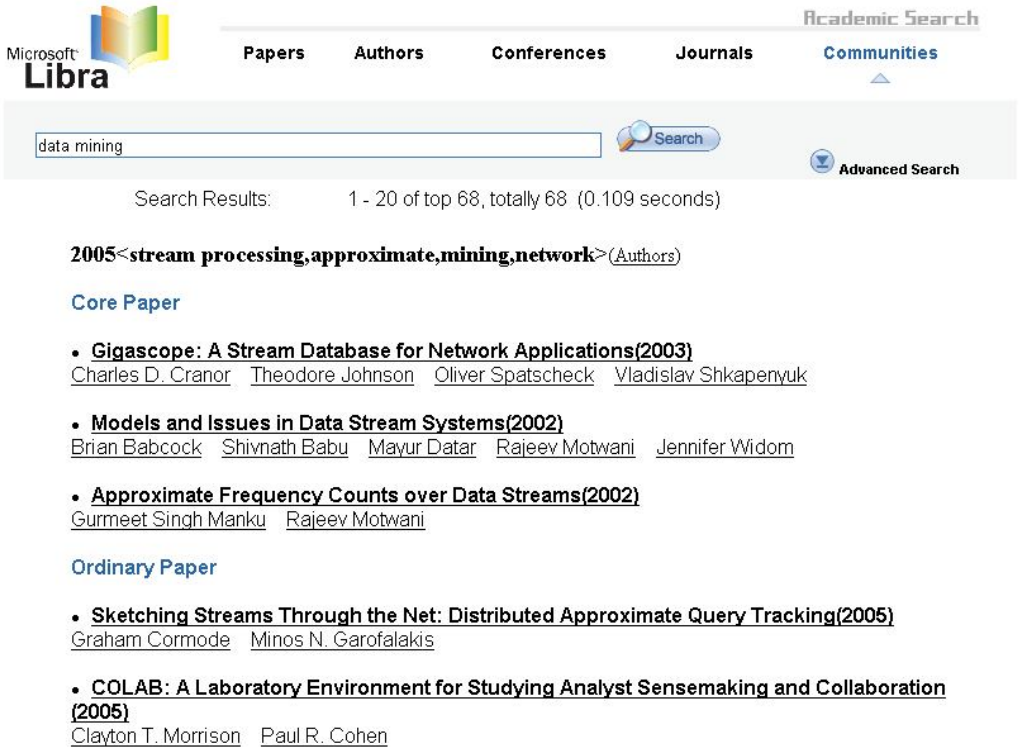
[3] DBLP http://dblp.uni-trier.de/

Obrázok 4: Informácie o autorovi v systéme DBconnect

5.4 Libra

Metadáta boli získané zo súborov PDF (Portable Document Format) (z ktorých možno získať aj citácie na iné publikácie) alebo webových databáz ACM Digital Library alebo IEEE Digital Library.

Portál zabezpečuje vyhľadávanie v metadátach na základe kľúčových slov v abstrakte publikácie, mena autora, názvu konferencie, názvu časopisu. Okrem toho poskytuje aj vyhľadávanie v komunitách. Komunity sa vyhľadávajú podľa kľúčových slov a roku.



Obrázok 5: Zobrazenie hlavných publikácií na zadanú tému systéme Libra

Výhodou je vyhľadávanie komunít podľa kľúčových slov a zobrazenie relevantných publikácií.

5.5 OpenAcademia

OpenAcademia je úložisko metadát pre vedecké komunity. Cieľom projektu je umožniť vedcom efektívnejšie zbierať, organizovať a rozširovať publikácie s využitím nástrojov webu so sémantikou. OpenAcademia je softvér, ktorý umožňuje, aby bol spustený lokálne a použitý pre spojenie s inými údajovými zdrojmi po svete. Bibliografické záznamy BibTeX sa konvertujú na formát, ktorý sa používa na rozširovanie správ (RSS).

Umožňuje iba manuálne vytváranie skupín.

5.6 Rozšírenie portálu ACM o značky

Autori [Freyne *et al.* 2007] navrhli prístup k informáciám s prvkami sociálneho filtrovania, ktorý používa znalosti komunity získané vyhľadávaním pre podporu navigácie a opačne. Vytvorili systém založený na výsledkoch vyhľadávača portálu ACM, ktorý pridáva sociálne značky (obrázok 6). Ak používateľ vyhľadáva, systém získa množinu výsledkov z portálu ACM a usporiada výsledky na základe anotácií, napríklad tým, že uprednostní články označené komunitou.

Obrázok 6: Vyhľadávanie obohatené o sociálne značky

Ďalej systém zobrazuje indikátory a anotácie pri výsledkoch. Spolu je pridaných päť ikon: Prvou zľava je percentuálna relevancia, ktorá znamená koľko krát bol tento výsledok vybraný komunitou použitím tohto dopytu, Druhou je ikona súvisiacich výsledkov, po ukázaní na túto ikonu sa zobrazia súvisiace dopyty pre tento výsledok, čím je používateľ informovaný o podobných témach ku ktorým môže článok prináležať. Systém vyhľadá aj články, ktoré neobsahujú kľúčové slová v dopyte sú ale relevantné, komunitou odporúčané pre tieto kľúčové slová. Treťou ikonou je aktuálnosť, čas ktorý uplynul od posledného vyhľadávania alebo prehľadávania pre túto publikáciu. Výsledky, ktoré neboli navštívené často nemusia byť také užitočné. Štvrtou ikonou sú stopy, ktoré znamenajú relatívnu popularitu článku pre dopyt. Piata ikona znázorňuje

prítomnosť anotácií. Anotované články obsahujú používateľom pridanú informáciu, ktorú používateľ, ktorý ju pridal považoval za relevantnú.

5.7 Zhodnotenie

Predstavili sme niekoľko existujúcich portálov, ktoré umožňujú správu metadát v oblasti publikačnej činnosti, pričom sme sa zamerali na možnosti vytvárania komunít. Niektoré z analyzovaných portálov umožňujú len manuálne vytvorenie komunít a iba portál Libra umožňuje objavovanie komunít. V tabuľke 2 sa nachádza porovnanie portálov. ArnetMiner a OpenAcademia sú portály, ktoré používajú nástroje webu so sémantikou a reprezentáciu metadát ontológiou. Portál ArnetMiner poskytuje prístup k metadátam jednotlivo pomocou webovej služby, nie je možný export celej ontológie. OpenAcademia poskytuje prístup do ontologického úložiska, metadát je v tomto portáli pomerne málo, pretože ontológia je napĺňaná manuálne používateľmi.

	Vytvorenie komunít	**Hľadanie spojení medzi autormi**	**Hľadanie autorít**	**Reprezetnácia metadát**
ArnetMiner	nie	áno	áno	ontológia
ACM (rozšírenie)	nie	nie	nie	relačná DB
Bibsonomy	manuálne	nie	nie	relačná DB
CiteSeer	nie	nie	nie	relačná DB
DBConnect	nie	áno	áno	relačná DB
Libra	áno	nie	áno	relačná DB
OpenAcademia	manuálne	áno	nie	ontológia

Tabuľka 2: Zhodnotenie portálov z hľadiska poskytovaných funkcií

6 Návrh metódy pre zdieľanie informácií o publikáciách

V tejto kapitole uvádzame metódu pre zdieľanie informácií, ktorá definuje postup od získania metadát, cez ich prípravu, analýzu a spracovanie a nakoniec prezentáciu portálom.

1. **Získanie metadát -** Primárnym zdrojom metadát je web. Na webe sú dostupné predspracované množiny metadát alebo je ich možné získať z webového sídla vydavateľa publikácií. Podrobnejšie v časti 7.4.

2. **Predspracovanie metadát –** úlohou predspracovania je prevod získaných metadát do formátu ontológie. Prevod pozostáva z priradenia identifikátorov zdrojov, prepojenia publikácií identifikovanými referenciami a úpravu mien autorov do jednotného tvaru. V rámci predspracovania sú vytvorené matice *autor-téma* a *spoluautor*, ktoré sú použité v analýze metadát. Podrobnejšie v časti 7.5.

3. **Analýza metadát, objavovanie komunít –** štandardné metódy objavovania komunít založené na metrikách (ako napríklad *GN algoritmus*, časť 4.2) potrebujú poznať metriku pre porovnanie inštancií, a pritom nie je zaručené, že objavia dobré komunity. Na rozdiel od nich sa štatistické metódy naučia štruktúru na základe údajov (angl. unsupervised learning).

   Prvou použitou metódou pre objavovanie komunít je štatistická technika (časť 4.3). Pôvodne bola určená na vyhľadávanie tém v dokumentoch skladajúcich sa zo slov. Túto techniku aplikujeme na objavovanie komunít autorov na základe spoločných záujmov odvodených od autorových publikácií. Rozšírenú metódu o uvažovanie odkazov (časť 4.4) pri hľadaní komunít autorov nemožno použiť pretože, predpokladom metódy je zviazanosť oboch matíc na publikácie.

   Druhá metóda používa pre objavovanie komunít graf, v ktorom hrany medzi autormi predstavujú vzťah spoluautorstva (časť 4.5).

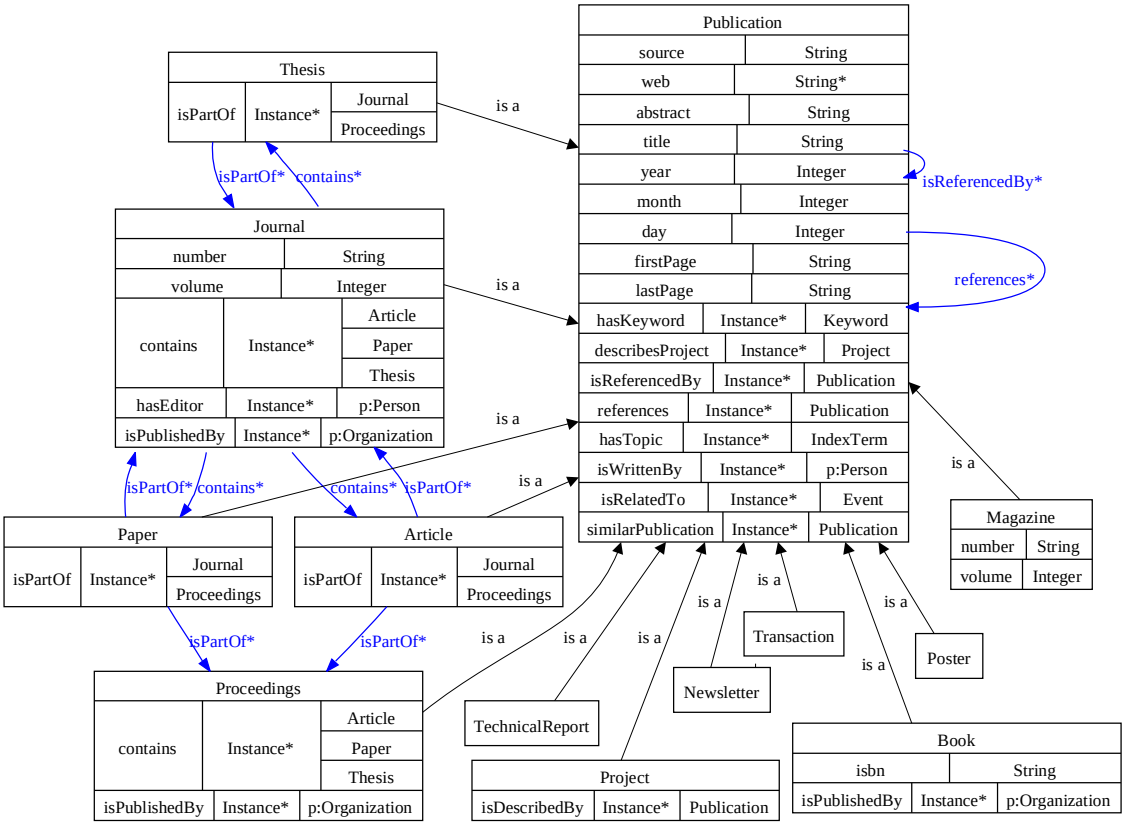
4. **Prezentácia portálom –** Zvolili sme dva režimy zobrazenia: objavené komunity metadáta publikácie a autora zobrazujeme v tabuľkách, spoluautorov vybraného autora zobrazujeme ako graf (pozri časť 9.2).

7 Získanie a predspracovanie metadát o publikáciách

V tejto časti prezentujeme navrhované rozšírenia ontológie vychádzajúc z cieľov práce a analýzy štruktúry metadát pre oblasť publikačnej činnosti z projektu MAPEKUS[4] a SWRC [Sure *et al.* 2005], ktoré boli zapísané v štandardnom formáte pre reprezentáciu ontológií OWL. Rozšírenia sa týkajú reprezentácie komunít a ohodnotenia autorov a publikácií.

7.1 Štruktúra metadát – projekt MAPEKUS

Tím výskumníkov z Ústavu informatiky a softvérového inžinierstva, FIIT STU, Bratislava pracoval na projekte MAPEKUS – Modelovanie a získavanie, spracovanie a využívanie znalostí o konaní používateľa v hyperpriestore internetu. V rámci neho bola vytvorená štruktúra metadát pre oblasť vedeckej publikačnej činnosti (Obrázok 7).



Obrázok 7: Štruktúra metadát pre oblasť publikačnej činnosti vytvorená v rámci projektu MAPEKUS

[4] Projekt MAPEKUS http://mapekus.fiit.stuba.sk/

Obrázok 7 znázorňuje štruktúru metadát, ktorá obsahuje základné atribúty publikácie ako *názov*, *dátum vydania*, a *autorov*, ktoré postačujú na jej identifikáciu referenciou. Okrem toho obsahuje *kľúčové slová* a *témy* týkajúce sa klasifikácie publikácie, zoznam referencií a rozširujúce informácie ako napríklad *abstrakt* a *odkaz na web*. Tento návrh štruktúry treba rozšíriť o hodnotenie publikácií a o štruktúru pre reprezentovanie komunity, aby sme ju mohli využiť pre ciele tejto práce.

7.2 Štruktúra metadát – projekt SWRC

SWRC[5] (Semantic Web for Research Communities) je ontológia pre modelovanie entít vedeckých komunít ako osoby, organizácie, publikácie (bibliografické metadáta) a ich vzťahov.

Štruktúra triedy publikácií vychádza z formátu metadát pre BibTeX[6] obsahuje triedy ako napríklad *článok*, *kniha*, *zborník*, *časopis*, *manuál*, *diplomová práca* a iné. Okrem základných vzťahov ako *autor* (tvorca) publikácie, *identifikátor*, *formát*, *nadpis*, *zdroj* možno definovať odkazy na iné publikácie (citácie), odkazy na súvisiaci projekt, organizáciu, a témy publikácie.

V rámci projektu SWRC bola vytvorená ontológia pre reprezentovanie komunít s názvom COIN (Communities of Interest Network). Hlavnou entitou v tejto ontológii je *Komunita*, ktorá má dátové vlastnosti *názov*, *popis komunity*, *stav* (formovaná, činná, uzatvorená), a odkazy na súvisiace komunity. Ontológia COIN zodpovedá jednoduchému rozšíreniu bez väzobnej entity členstvo (bližšie v kapitole 7.3).

[5] Semantic Web for Research Communities http://ontoware.org/projects/swrc/

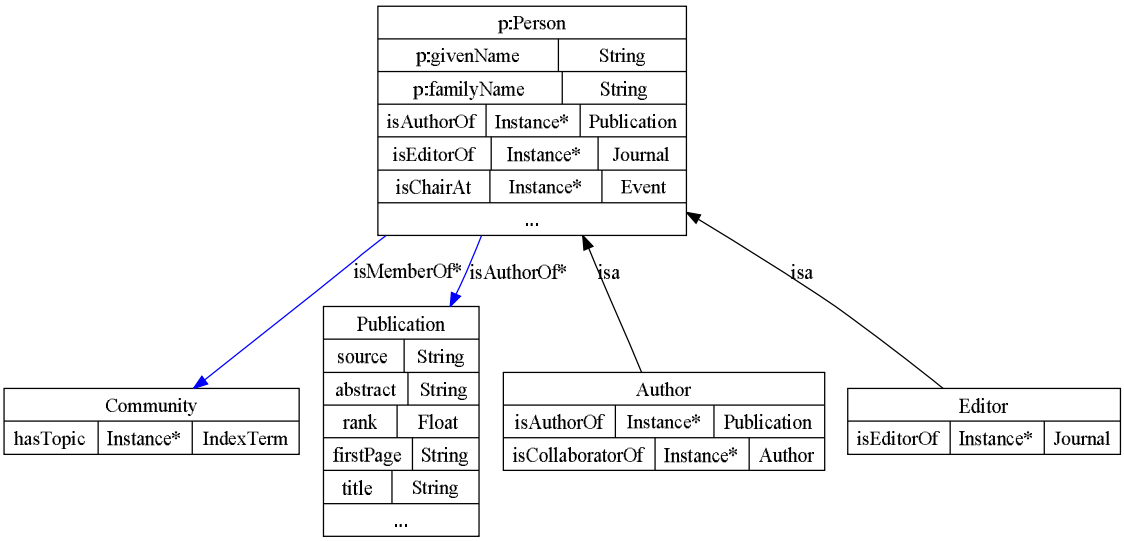
[6] BibTeX http://www.bibtex.org/

7.3 Návrh rozšírenia štruktúry metadát

Pre účely objavenia komunít výskumníkov sme rozšírili štruktúru metadát opísanú v predchádzajúcich častiach o ďalšie entity.

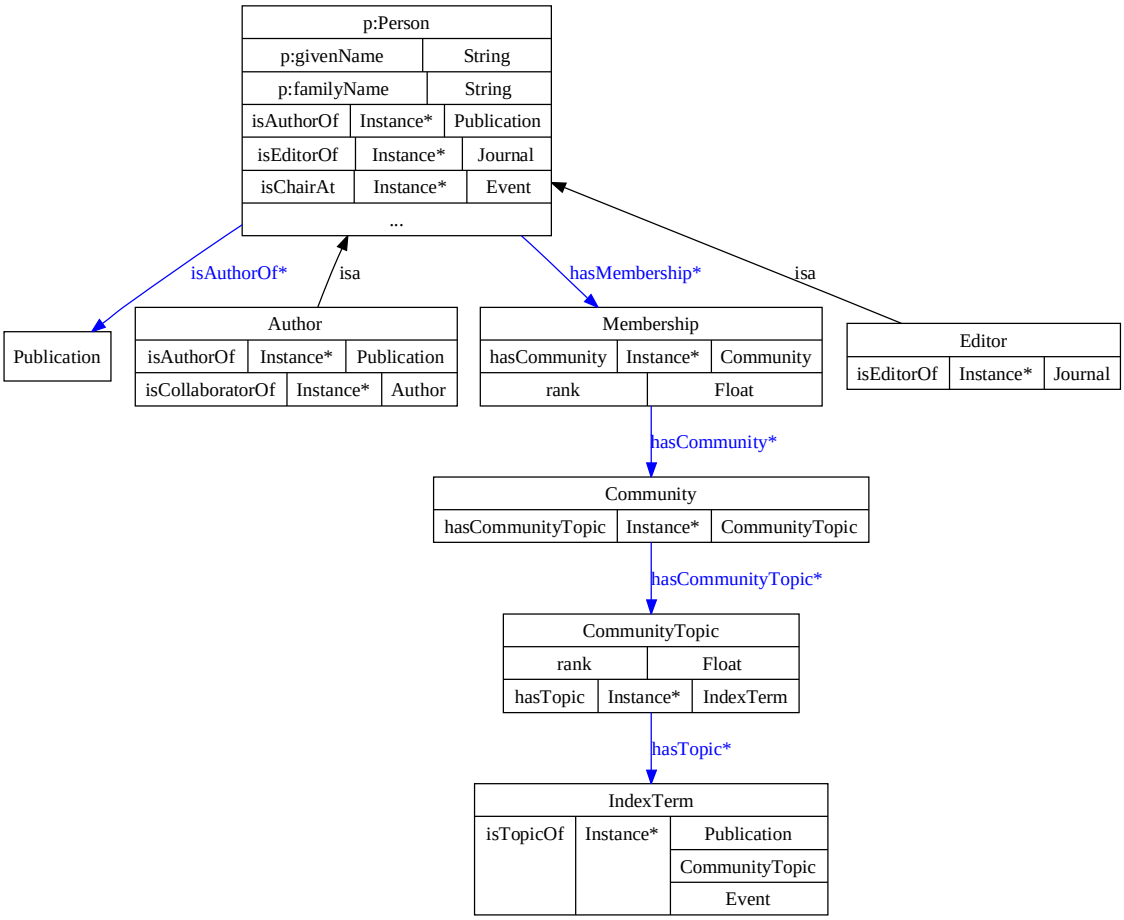
- Vytvorili sme triedu **Community** s dátovou vlastnosťou **hasTopic** (multiple IndexTerm)
- Publikácii sme pridali atribút hodnotenia **rank** typu xsd:double
- Triede party:Person sme pridali atribút **isMemberOf** (multiple Community)

Rozšírenie, ktoré znázorňuje Obrázok 8 má výhody aj nevýhody. Výhodou je jednoduchosť reprezentácie, ktorá nepoužíva väzobné entity. V tomto zjednodušenom rozšírení sa predpokladá, že témy vybranej komunity majú rovnakú váhu a príslušnosť autora je vyjadrená vzťahom členstva bez ďalších atribútov, ako napríklad autorove preferencie ku komunite.



Obrázok 8: Štruktúra metadát po pridaní triedy Community

Relácia sa reprezentuje samostatnou dátovou entitou, ak má relácia vlastné atribúty [Šešera *et al.* 2000], napríklad preferenciu autora ku komunite. Z toho vyplýva, že je nutné vytvoriť toľko inštancií väzobnej entity **Membership,** koľko je párov Author-Community. Tiež je nutné vytvoriť toľko inštancií väzobnej entity **CommunityTopic** koľko je párov Community-IndexTerm, ak požadujeme, aby každá téma mala rozdielne váhy pre každú komunitu. Potom bude štruktúra metadát vyzerať tak, ako znázorňuje Obrázok 9.



Obrázok 9: Rozšírenie modelu o väzobné entity Membership a CommunityTopic

Pre úplnosť vyššie uvedeného modelu, je možné ho doplniť aj o inverzné relácie (v smere zdola nahor), avšak za cenu zvýšenej zložitosti pri udržiavaní inštancií.

7.4 Získanie metadát

Okrem navrhnutia štruktúry ontológie je potrebné naplniť ontológiu inštanciami. Primárnym zdrojom metadát je web. Vydavatelia publikácií majú na svojich webových sídlach metadáta publikácií. Stránky niektorých vydavateľstiev ponúkajú okrem nadpisu publikácie, zoznamu autorov a abstraktu aj zoznam referencií.

Na webe je dostupných niekoľko množín metadát v rôznych formátoch. Cieľom je konverzia do navrhnutej reprezentácie ontológie vo formáte RDF/OWL.

Pre účely projektu sme vyhľadali niekoľko rozsiahlych zdrojov metadát publikácií: ACM[7], CiteSeer[8], Cora [McCallum 2001] a DBLP[9]. Bolo potrebné vybrať vhodný zdroj metadát. V tabuľke 1 sme porovnali vlastnosti zdrojov metadát z hľadiska niekoľkých pre naše potreby významných kritérií. S ohľadom na korektnosť metadát a odhadnutej zložitosti spracovania sme sa rozhodovali medzi viacerými možnosťami.

	ACM*	CiteSeer	Cora	DBLP
Verzia/dátum	2008	2008	2001	2008
Kategorizácia	+	–	+	–
Kľúčové slová/abstrakt	+	+	–	–
Konferencia/Zborník	+	+	–	+
Odkazy DOI	+	+	–	–
Citovanie (bibliografické odkazy)	+	+	+	+
Formát pôvodných údajov	OWL	XML	TXT	XML
Rozlíšené referencie	–	+	+	+

* dáta z projektu MAPEKUS

Tabuľka 1: Porovnanie metadát pre ontológie

Odkazy *Document Object Identifier*[10] (DOI) predstavujú jedinečný identifikátor publikácií, aktuálne v procese štandardizácie ISO. Aktivovaním DOI odkazu

[7] ACM Digital Library http://portal.acm.org/dl.cfm

[8] CiteSeer http://citeseer.ist.psu.edu/

[9] DBLP http://dblp.uni-trier.de/

[10] Document Object Identifier http://www.doi.org/

vo webovom prehliadači dôjde k presmerovaniu na adresu webového sídla vydavateľa, kde možno nájsť metadáta publikácie a objednať si sprístupnenie textu celej publikácie.

V portáli ACM sú údaje o publikáciách dostupné len vo forme HTML stránok. Obaľovač (angl. wrapper) vytvorený v rámci projektu MAPEKUS na základe spoločnej štruktúry generovaných stránok vyberá špecifikované metadáta.

V rámci riešenia sme analyzovali štyri zdroje údajov:

1. **Digitálna knižnica *ACM*.** Publikácie mali referencie, ale neboli cez ne vzájomne poprepájané, pretože boli vytvorené strojovým rozpoznávaním textu publikácie.
2. **Databáza *CiteSeer.*** [Bollacker 2001]. Predstavje rozsiahlu množinu avšak mená autorov majú vynechané písmená s diakritikou a obsahujú preklepy.
3. **Databáza *Cora*.** Údaje obsahujú klasifikáciu, ktorá bola vytvorená klasifikátorom do predvolených kategórií. Každá publikácia bola priradená práve do jednej kategórie. Mená autorov boli už upravené do jednotného tvaru.
4. **Databáza *DBLP*.** Pre účely naplnenia ontológie publikácií bol vyvinutý nástroj pre konverziu z pôvodného formátu. Nástroj okrem mapovania elementov zo vstupného formátu vykonáva úpravu na jednotný tvar a priraďovanie jednoznačných identifikátorov zdrojom.

Nakoniec pre experimenty v rámci tejto práce sme použili množinu Cora, pretože metadáta obsahovali referencie a klasifikáciu do 70 tém. Na druhú stranu je z hľadiska údajov táto množina je neaktuálna, pretože najnovšie metadáta sú z roku 1999. Na konverziu metadát z textového formátu do ontológie sme vytvorili samostatný modul.

7.5 Predspracovanie metadát

Postupnosť prevodu množiny Cora pozostáva z týchto krokov:

1. Priradenie identifikátora publikácii.
2. Prepojenie publikáciami cez referencie.
3. Vytvorenie zoznamu autorov a úprava mien na spoločný tvar.
4. Priradenie názvu, témy a roku.

Medzivýsledky prevodu metadát sa ukladanú do relačnej databázy. Výsledné predspracované metadáta sú zapísané vo formáte ontológie OWL. Správnosť naplnenej ontológie možno skontrolovať jej otvorením v editore *Protégé*.

Prepojenie publikácie cez referencie si vyžaduje rozpoznanie položiek citácií. Cieľom je vytvoriť prepojenia medzi citujúcou a citovanou publikáciou.

Skupina prístupov pre rozpoznávanie referencií v publikáciách založených na metódach strojového učenia má zvyčajne podstatne vyššiu presnosť rozpoznávania ako prístupy založené na pravidlách (pre porovnanie metód pozri napríklad [Kiat 2005]). Metóda *Conditional Random Fields* (CRF) [Lafferty et al. 2001], dosahuje pri rozpoznávaní položiek citácií vyššiu presnosť ako *Hidden Markov Model* (HMM) alebo *Maximum entropy method* (MEM).

Pre rozpoznávanie v rámci tohto projektu sme použili model z nástroja ParsCit[11], ktorý používa knižnicu CRF++. Rozpoznávanie dokáže pracovať len s referenciami bez diakritiky.

Po rozpoznaní položiek citácií nasleduje porovnávanie metadát publikácií s rozpoznanými citáciami a vytvorenie prepojení na referencované metadáta.

[11] ParsCit http://wing.comp.nus.edu.sg/parsCit/

8 Analýza a spracovanie metadát

Z predspracovaných metadát potrebujeme získať graf vzájomných citácií pre výpočet hodnotenia publikácií a graf spoluautorstva.

Graf vzájomných citácií

Formálna komunikácia je zachytená pomocou citovania – odkazovania na iné publikácie. Graf citácií je graf, v ktorom vrcholy reprezentujú publikácie a hrany reprezentujú citácie. Citovaná publikácia predstavuje zdroj znalostí pre citujúcu publikáciu, ktorá zhromažďuje znalosti. Hlavné rozdiely grafu citácií od iných typov grafov sú [Leicht & Newman 2007]:

1. Uzly majú asymetrické vzťahy, dôsledkom sú orientované hrany grafe.
2. Graf sa rozvíja v čase a dokumenty je možné iba pridávať na rozdiel napríklad od webu, v ktorom dokumenty môžu byť odobrané alebo po pridaní môžu zmeniť umiestnenie.
3. Graf je acyklický, čo vyplýva z predchádzajúcich dvoch tvrdení. Publikácia môže citovať len publikácie z grafu. Inak povedané v grafe už musí existovať uzol, ktorý reprezentuje citovanú publikáciu.

Graf spoluautorstva

Graf spoluautorstva je špeciálnym prípadom sociálnej siete, v ktorej uzly reprezentujú autorov a hrany znamenajú spoluprácu medzi autormi. Hrana, ktorá spája dvoch autorov vyjadruje fakt, že títo autori sú alebo boli kolegovia. Publikovali spolu jeden alebo viacero publikácií ako výsledok spoločného výskumu. Siete spoluautorstva môžu tiež vyjadrovať intenzitu spolupráce vyjadrenú váhou hrany. Pri priraďovaní váhy spolupráce možno brať do úvahy počet spoluautorov alebo počet spoločne napísaných publikácií [Ježek *et al.* 2008].

8.1 Metóda identifikácie záujmov autora

Podobné záujmy autora sa identifikujú na základe publikácií, ktorých je autorom alebo spoluautorom. Predpokladom je, že každá publikácia má určenú aspoň jednu tému. Hodnotenie vybranej publikácie vypočítame algoritmom PageRank, pretože toto hodnotenie vyjadruje celkovú autoritu autora (pozri kapitolu 3).

Ohodnotenie uzlov grafu citácií algoritmom hodnotenia autoritatívnosti publikácií

Vstup: Ontológia publikácií.

Vstupná podmienka: Publikácia musí referencovať alebo byť referencovaná, aby mohla byť ohodnotená.

Výstup: Každá publikácia má priradené hodnotenie.

Výstupná podmienka: Súčet hodnotení všetkých publikácií je rovný jedna.

Algoritmus:

1. Vytvorenie grafu:
    a. Získanie identifikátora každej publikácie z ontológie.
    b. Pre každú publikáciu získanie citovaných publikácií z ontológie.
2. Výpočet – iteratívny výpočet vybraným algoritmom na vytvorenom grafe.
3. Uloženie výsledkov – pre každú publikáciu zapísanie jej hodnotenia do ontológie.

Ohodnotenie autorov na základe ohodnotenia publikácií

Vstup: Ontológia z domény publikačnej činnosti

Vstupné podmienky:

1. Každá publikácia musí mať aspoň jednu tému
2. Aby bol autor ohodnotený musí mať aspoň jednu publikáciu ohodnotenú.

Výstup: Matica Autor-Téma s prvkami vyjadrujúcimi preferenciu autora k téme.

Výstupné podmienky: Súčet hodnotení všetkých tém všetkých autorov je rovný jedna.

Algoritmus:

Získanie identifikátorov všetkých publikácií z ontologického úložiska

Pre každú publikáciu p:

- Získanie identifikátorov tém t publikácie p
- Získanie identifikátorov hodnotenia h publikácie p
- Získanie identifikátorov autorov a publikácie p
- Výpočet hodnotenia r vzhľadom na spoluautorstvo a počet tém: $r = (h/|a|)/|t|$
- Pre každého autora a:
  - Zisti témy $t(a)$ autora
  - Pre každú tému $t(a)$:
    - Pripočítanie r k hodnoteniu autor-téma

Poradie autorov v referencii môže vyjadrovať, že prvý uvedený autor má vyšší podiel na tvorbe publikácie, avšak táto skutočnosť by musela byť zohľadnená aj v ontológii, kde by muselo byť špecifikované poradie autorov.

8.2 Vytvorenie skupín výskumníkov

Skupiny vytvárame na základe podobných záujmov autorov identifikovaných metódou identifikácie záujmov autora.

Ako prvú metódu hľadania komunít sme použili pravdepodobnostnú latentnú sémantickú analýzu (kapitola 4.3), ktorá analyzuje maticu vytvorenú v predchádzajúcom kroku. Počet komunít, ktoré majú byť objavené je nutné špecifikovať na začiatku algoritmu. Mal by byť menší ako obidva rozmery matice, aby došlo k redukcii rozmernosti analyzovaných dát.

Vstup: matica M vyjadrujúca záujem autora o tému, počet komunít.

Vstupné podmienky: každý autor má aspoň jednu tému s nenulovým hodnotením.

Výstup: matica priradenia autorov do komunít P(z|a) a matica rozdelenia tém v komunitách P(t|z).

Výstupné podmienky: každý autor je priradený aspoň do jednej komunity.

Algoritmus:

Inicializuj matice P(z|a), P(t|z) náhodnými hodnotami a normalizuj riadky.

Inicializuj matice P'(z|a), P'(t|z) nulovými prvkami.

Pokiaľ nie je dosiahnutá presnosť alebo maximálny počet iterácií:

```
Pre každého autora a:
    Pre každú tému t:
        menovateľ ← 0
        Pre každú komunitu z:
            menovateľ ← menovateľ + P(z|a) * P(t|z)
        Pre každú komunitu z:
            Ak (menovateľ = 0)
                P(z|a, t) ← 0
            Inak
                P(z|a, t) ← (P(z|a) * P(t|z)) / menovateľ
            čitateľ ← M(a, t) * P(z|a, t)
            P‘(t|z) ← P‘(t|z) + čitateľ
            P‘(z|a) ← P‘(z|a) + čitateľ
Pre každú komunitu z:
    suma ← 0
    Pre každú tému t:
        suma ← suma + P’(t|z)
    Pre každú tému t:
        P(t|z) ← P’(t|z) / suma
        P‘(t|z) ← 0
Pre každého autora a:
    suma ← 0
    Pre každú tému t:
        suma ← suma + M(a,t)
    Pre každú tému z:
        P(z|a) ← P’(z|a) / suma
        P‘(z|a) ← 0
```

E krok – výpočet očakávanej pravdepodobnosti

M krok – aktualizácia parametrov modelu

Ako druhú metódu hľadania komunít sme použili pravdepodobnostný model pre homogénny graf spoluautorov (kapitola 4.5).

Vstup: matica susednosti M vyjadrujúca spoluautorstvo.
Vstupné podmienky: prvky na diagonále matice spoluautorstva sú rovné jedna.
Výstup: vektor π_r vyjadruje dôležitosť komunity r v rámci všetkých komunít, a matica $\beta_{r,i}$ vyjadruje príslušnosť autora i ku komunite r.
Výstupné podmienky: každý autor je priradený aspoň do jednej komunity.

Algoritmus:
Inicializuj vektor π_c a maticu $\beta_{c,i}$ náhodnými hodnotami a normalizuj vektor a riadky matice
Inicializuj vektor π'_c a maticu $\beta'_{c,i}$ nulovými prvkami.
Pokiaľ nie je dosiahnutá presnosť alebo maximálny počet iterácií:

```
Pre každého autora i:
    Pre každého autora j:
        menovateľ ← 0
        Pre každú komunitu c:
            menovateľ ← menovateľ + π_c * β_{c,i} * β_{c,j}
        Pre každú komunitu c:
            Ak (menovateľ = 0)
                P(e_{i,j}) ← 0
            Inak
                P(e_{i,j}) ← (π_c * β_{c,i} * β_{c,j}) / menovateľ
            π'_c ← π'_c + P(e_{i,j})
            β'_{c,i} ← β'_{c,i} + P(e_{i,j})
```

E krok – výpočet očakávanej pravdepodobnosti

```
suma ← 0
Pre každú komunitu c:
    suma ← suma + π'_c
Pre každú komunitu c:
    π_c ← π'_c / suma
Pre každú komunitu c:
    suma ← 0
    Pre každého autora i:
        suma ← suma + β'_{c,i}
    Pre každého autora i:
        β_{c,i} ← β'_{c,i} / suma
        β'_{c,i} ← 0
```

M krok – aktualizácia parametrov modelu

9 Overenie riešenia – portál pre zdieľanie výsledkov publikačnej činnosti

9.1 Vývoj portálu

Hlavným znakom portálu je vstupná stránka s mnohými odkazmi na iné stránky. Portál prezentuje objavené komunity autorov. Doplnkovou funkciou portálu je vyhľadávanie podľa mena autora a kľúčových slov v nadpise publikácie. Webové stránky portálu sú vytvárané dynamicky na základe dopytu klienta (klientom je napríklad webový prehliadač používateľa).

Portál poskytuje zobrazenie zoznamu autorov v komunitách. Autori sú usporiadaní podľa hodnotenia autorov. Hodnotenie autorov je vytvorené na základe hodnotenia publikácií. V tabuľke sa vedľa každého autora nachádza stĺpcový graf spolu s hodnotením autora v rozsahu 1-10.

Z tabuľky možno aktivovať dva typy odkazov:

- Poradové číslo komunity – zobrazí sa zoznam autorov vo vybranej komunite usporiadaný zostupne podľa hodnotenia autora.
- Meno autora – zobrazia sa spoluautori spolu so zoznamom publikácií autora.

Aktivovaním odkazu na autora sa zobrazí tabuľka publikácií autora. Tabuľka obsahuje metadáta publikácie ako rok vydania, názov a hodnotenie. Ostatné metadáta publikácie sa zobrazia nasledovaním odkazu, ktorý tvorí nadpis publikácie.

Na implementáciu portálu sme použili jazyk *Java* integrované vývojové prostredie *Eclipse* s rozširujúcimi modulmi pre prácu s manažmentom verzií *Subversion* (JavaSVN, Subclipse) a nástrojom pre manažment softvérového projektu *Maven*.

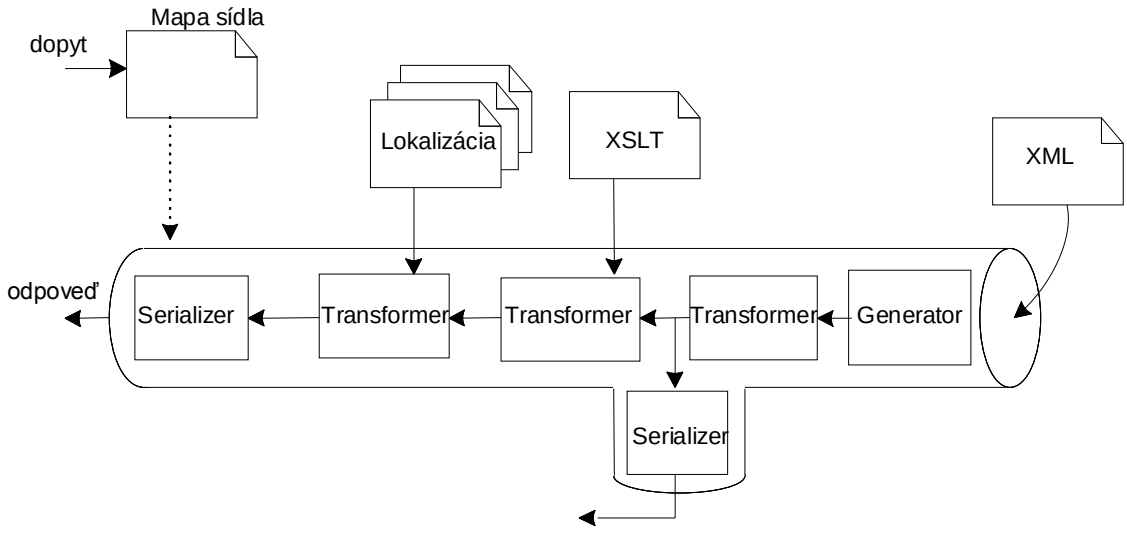
Implementácia algoritmu hodnotenia PageRank bola použitá z knižnice *JUNG* (Java Universal Network/Graph library). Na reprezentáciu riedkych matíc bola použitá knižnica *COLT*. Vykresľovanie grafu na strane klienta (Java applet) zabezpečuje knižnica *Prefuse*.

Apache Tomcat bol zvolený ako server pre umiestnenie portálu a ontologického úložiska. Portál obsahuje knižnicu rámca pre tvorbu webových portálov Apache Cocoon.

Apache Cocoon je založený na architektúre dátovodov a filtrov. Údaje sú spracovávané postupne ako prechádzajú dátovodom. Filtre sú troch typov: generátory, tranformery a serializery. Každý dátovod začína generátorom, za ktorým nasleduje určitý počet transformerov a ukončuje sa serializerom. Generátor má za úlohu zo vstupného zdroja dát (napríklad súbor) vytvoriť vnútornú reprezentáciu dokumentu ako postupnosť udalostí SAX (Simple API for XML). Transformery potom spracovávajú takto reprezentovaný dokument. Serializer túto vnútornú reprezentáciu zapíše do špecifikovaného výstupného súboru (napríklad HTML, PDF, JPEG).

Obrázok 10 znázorňuje postup spracovania rámcom Apache Cocoon. Na základe požiadavky klienta, server vyhľadá dátovod v mape sídla. Server potom začne spracovávať dopyt v smere od generátora po serializer. Na obrázku odbočka z dátovodu umožňuje v určitom bode zapísať obsah vnútornej reprezentácie, čo možno využiť pri detekcii chýb a pri exporte do súboru XML.

Transformer umožňuje vkladať (rekurzívne) obsah iných dátovodov definovaných v mape sídla, čo sme využili pri rozložení spracovania.



Obrázok 10: Spracovanie požiadavky klienta - vzor dátovody a filtre

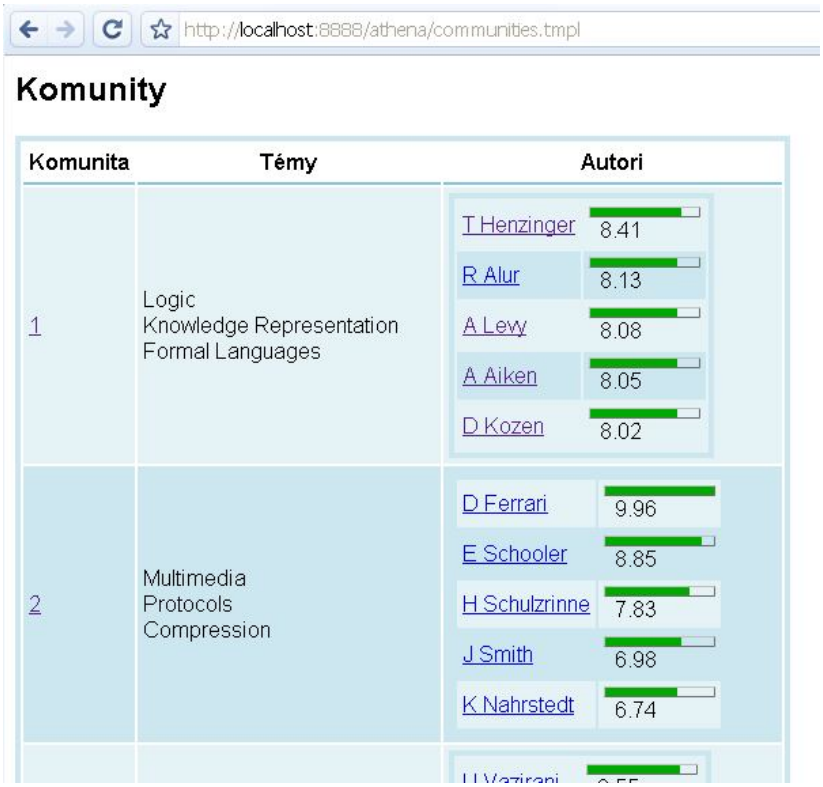
9.2 Spôsob prezentácie v portáli

Portál poskytuje dva režimy zobrazenia:

- formou tabuľky
- formou grafu.

9.2.1 Zobrazenie formou tabuľky

Obrázok 11 znázorňuje tabuľku objavených komunít autorov v portáli. Pre každú komunitu sú zobrazené témy, ktoré túto komunitu charakterizujú spolu s niekoľkými autormi s najvyšším hodnotením.

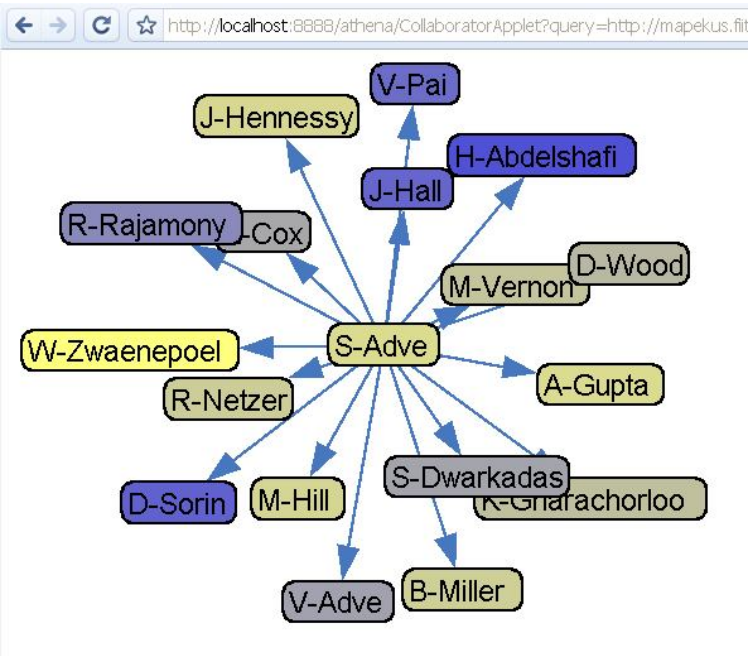


Obrázok 11: Zobrazenie komunít autorov spolu s témami

9.2.2 Zobrazenie formou grafu

Obrázok 12 znázorňuje *Java Applet*, ktorým sa zobrazuje graf spoluautorov. Farba uzla vyjadruje jeho autoritu od modrej pre nízku autoritu po žltú pre vysokú autoritu.

Aktivovaním uzla ľavým tlačidlom sa možno presúvať po sieti spoluautorstva. Držaním ľavého tlačidla možno presúvať graf v okne prehliadača. Aktivovaním uzla stredným tlačidlom sa zobrazia detaily tohto autora. Stlačením pravého tlačidla sa graf vycentruje v okne prehliadača. Držaním pravého tlačidla a pohybom vo vertikálnom smere možno graf približovať a vzďaľovať.



Obrázok 12: Zobrazenie grafu spoluautorstva

9.3 Experimenty s malou vzorkou údajov

V tejto časti opisujeme postup hľadania komunít na malej testovacej vzorke. Pre názornosť sú údaje prezentované ako matice.

Matica Autor–Publikácia

	pub1	pub2	pub3	pub4
http://...publication#authorChotirat	1			1
http://...publication#authorCzerwon				1
http://...publication#authorDufourd		1		1
http://...publication#authorMcMahon	1	1	1	
http://...publication#authorHsiao	1		1	

Tabuľka 3: Matica autor–publikácia

Matica vzájomných citácií

Publikácia v riadku cituje publikáciu v stĺpci.

	pub1	pub2	pub3	pub4
http://...publication#pub1				
http://...publication#pub2	1			
http://...publication#pub3	1	1		
http://...publication#pub4	1			

Tabuľka 4: Matica vzájomných citácií

Vypočítané hodnotenia pre každú publikáciu:

	PageRank
http://...publication#pub1	0.504431
http://...publication#pub2	0.206185
http://...publication#pub3	0.144692
http://...publication#pub4	0.144692

Tabuľka 5: Vypočítané hodnotenie pre publikácie

9.3.1 Matica Publikácia–Téma

Publikácia môže mať viacero tém

	t3_Internet	t4_Java	t2_Compilers	t1_Semantic_Web
http://... publication#pub1	1			1
http://...publication#pub2		1		
http://...publication#pub3			1	
http://...publication#pub4	1			

Tabuľka 6: Matica publikácia-téma

9.3.2 Matica Autor–Téma

Záujem autora o tému je vyjadrený maticou autor–téma. Matica bola vytvorená na základe hodnotenia PageRank vypočítaného pre každú publikáciu. Hodnotenie bolo rozdelené medzi autorov. V prípade viacerých tém publikácie sa hodnotenie ďalej rozdelilo tak, aby prispelo ku každej téme publikácie.

	t3_Internet	t4_Java	t2_Compilers	t1_Semantic_Web
http://... publication#authorChotirat	0.132302	0.000000	0.000000	0.084072
http://...publication#authorCzerwon	0.048231	0.000000	0.000000	0.000000
http://...publication#authorDufourd	0.048231	0.103093	0.000000	0.000000
http://...publication#authorMcMahon	0.084072	0.103093	0.072346	0.084072
http://...publication#authorHsiao	0.084072	0.000000	0.072346	0.084072

Tabuľka 7: Matica autor-téma

9.3.3 Rozdelenie autorov do komunít

Tabuľka 8 je znázorňuje rozdelenie autorov do komunít. Riadok tabuľky začína identifikátorom autora, nasledujú tri najpravdepodobnejšie skupiny pre tohto autora. Príslušnosť k skupine je vyjadrená číslom, za ktorým nasleduje pravdepodobnosť s akou autor prináleží do tejto skupiny.

http://... publication#authorChotirat	**3** 0.999991	**2** 0.000009	**1** 0.000000
http://... publication#authorCzerwon	**3** 0.999999	**1** 0.000001	**2** 0.000000
http://... publication#authorDufourd	**1** 0.999987	**3** 0.000013	**2** 0.000000
http://... publication#authorMcMahon	**2** 0.450974	**1** 0.425213	**3** 0.123813
http://... publication#authorHsiao	**2** 0.579835	**3** 0.420165	**1** 0.000000

Tabuľka 8: Rozdelenie autorov do komunít

Témy pre každú skupinu sú uvedené v ďalšej tabuľke.

9.3.4 Rozdelenie tém pre komunity

V tejto tabuľke každý riadok predstavuje témy pre komunitu. V riadku nasleduje niekoľko najpravdepodobnejších tém pre komunitu.

Z: 1	**t4_Java** 0.693248	**t3_Internet** 0.306752	**t1_Semantic_Web** 0.000000	**t2_Compilers** 0.000000
Z: 2	**t2_Compilers** 0.491492	**t1_Semantic_Web** 0.422971	**t3_Internet** 0.085531	**t4_Java** 0.000006
Z: 3	**t3_Internet** 0.687165	**t1_Semantic_Web** 0.312834	**t2_Compilers** 0.000000	**t4_Java** 0.000000

Tabuľka 9: Rozdelenie pravdepodobností pre témy komunít

9.4 Experimenty s rozsiahlou vzorkou údajov

Pre experimenty sme použili množinu Cora. Počet publikácií bol 11 474, počet autorov 11 481 a hierarchia tém reprezentovaná stromom mala 70 listových uzlov. Témy publikácií sú určené ich klasifikáciou, každá publikácia má danú práve jednu tému (na základe pôvodných metadát), čo je pre účely určenia záujmov autora postačujúce.

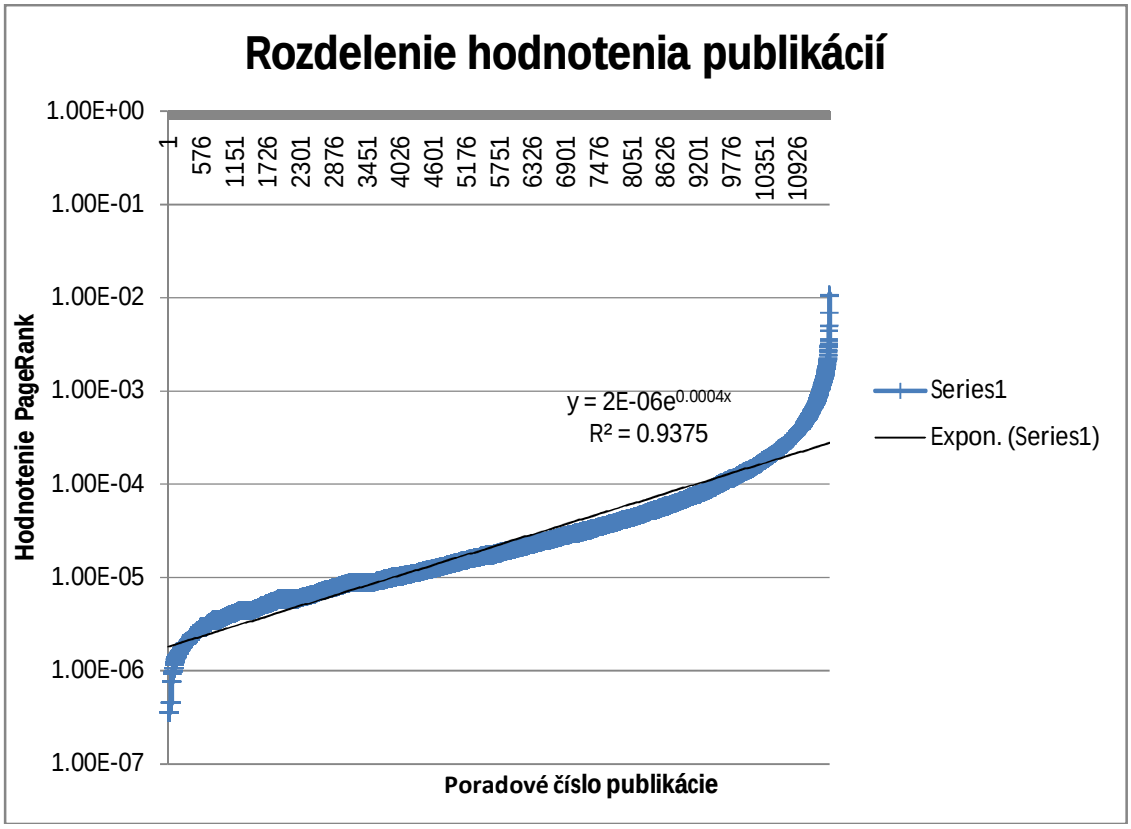
Prvá metóda (kapitola 4.3) vypočítala komunity výskumníkov. Počet komunít, ktoré majú objavené algoritmom bol nastavený na 32. Pri voľbe komunít vyššom ako menší z dvoch rozmerov matice sa stávalo že sa vytvárali komunity ktoré tvoril 1-2 autori. Výpočtová zložitosť algoritmu závisí lineárne od počtom komunít a celková zložitosť algoritmu je $\mathcal{O}(|a||t||z|)$, kde $|a|$ je počet autorov, $|t|$ je počet tém, $|z|$ je počet komunít.

Druhá metóda aj napriek tomu, že na malom grafe spoluautorov dávala dobré výsledky (kapitola 4.5) je výpočtovo náročná s kvadratickou zložitosťou od počtu autorov, a preto neboli vypočítané výsledky pre rozsiahlu množinu údajov. Celková zložitosť algoritmu je $\mathcal{O}(|a|^2|z|)$

Obrázok 13 znázorňuje hodnotenie publikácií usporiadané vzostupne. Z obrázku vyplýva, že usporiadané hodnotenia publikácií majú exponenciálny tvar, preto pre zobrazenie do lineárnej stupnice 1-10 je potrebné vypočítať logaritmus hodnotenia pre každú publikáciu ako

$$r'_p = 1 + 9 * \frac{\log(r_p) - \log(\min r_p)}{\log(\max r_p) - \log(\min r_p)}$$

Kde r'_p je transformované hodnotenie, r_p je hodnotenie PageRank publikácie p.



Obrázok 13: Rozdelenie hodnotenia publikácií v logaritmickej mierke

10 Zhodnotenie

Hlavným cieľom práce bol návrh metódy pre objavovanie komunít zo získaných metadát v oblasti publikačnej činnosti a návrh prototypu portálu pre ich prezentáciu s využitím nástrojov webu so sémantikou.

Na základe cieľov tejto práce sme počas vypracovania tohto projektu:

1. analyzovali štruktúru metadát a rozšírili sme ju tak, aby bolo možné reprezentovať skupiny výskumníkov.
2. získali niekoľko dostupných metadát pre oblasť publikačnej činnosti z webu a vytvorili softvér pre ich prevod do reprezentácie ontológiou.
3. preskúmali metódy pre vytvorenie skupín výskumníkov na základe metadát publikácií. Implementovali sme dve štatistické metódy hľadania komunít.
4. vytvorili prototyp portálu, ktorý používa reprezentáciu metadát ontológiou a umožňuje zobrazovanie objavených komunít autorov.

Tradičné metódy hľadania komunít, ktoré využívajú rozdeľujúci prístup sú pre grafy s veľkým počtom uzlov výpočtovo náročné, preto sme pre hľadanie komunít použili štatistické metódy. Výsledky potvrdzujú, že je možné získať komunity autorov na základe podobných záujmov s využitím grafu vzájomných citácií.

Prínosom práce je vytvorenie prehľadu o autoroch, ktorí tvoria komunity v nejakej oblasti výskumu.

Možné rozšírenie portálu môže predstavovať editáciu metadát cez webové rozhranie.

Použitá literatúra

[Bollacker *et al.* 1998] Bollacker, K., Lawrence, S., Giles, C. (1998). CiteSeer: an autonous Web agent for automatic retrieval and identification of interesting publications. *Proceedings of the second international conference on Autonomous agents* , 116–123. http://clgiles.ist.psu.edu/papers/Agents-1998-citeseer-agent.pdf

[Breslin *et al.* 2005] Breslin, J., Harth, A., Bojars, U., Decker, S. (2005). Towards Semantically-Interlinked Online Communities. *The 2nd European SemanticWeb Conference*, *LNCS 3532*, pp. 500–514. Heraklion, Greece. http://sw.deri.org/2004/12/sioc/index.pdf

[Capocci *et al.* 2005] Capocci, A., Servedio, V. D. P., Caldarelli, G., & Colaiori, F. (2005) Detecting communities in large networks. *Physica A: Statistical Mechanics and its Applications*. 352(2), 669–676. Elsevier. http://arxiv.org/pdf/cond-mat/0402499

[Chakrabarti *et al.* 1999] Chakrabarti, S., Dom, B., Kumar, S., Raghavan, P., Rajagopalan, S., Tomkins, A., Gibson, D., & Kleinberg, J. (1999). Mining the Web's link structure. *Computer*, 32(8), 60–67. http://www2.computer.org/portal/web/csdl/doi/10.1109/2.781636

[Cohn & Hofmann 2001] Cohn, D. and Hofmann, T. (2001) The Missing Link-A Probabilistic Model of Document Content and Hypertext Connectivity. Advances in Neural Information Processing Systems. pp. 430–436. MIT. http://www.cs.cmu.edu/Web/Groups/NIPS/00papers-pub-on-web/CohnHofmann.pdf

[Flake *et al.* 2004] Flake, G., Tsioutsiouliklis, K., & Zhukov, L. (2004). Methods for Mining Web Communities: Bibliometric, Spectral, and Flow. *Web Dynamics: Adapting to Change in Content, Size, Topology and Use.* http://www.inf.unibz.it/~ricci/SDB/papers/flake03.pdf

[Freyne *et al.* 2007] Freyne, J., Farzan, R., Brusilovsky, P., Smyth, B., Coyle, M. (2007) Collecting community wisdom: integrating social search & social navigation. Proceedings of the 12th international conference on Intelligent user interfaces. pp. 52–61. ACM Press New York, NY, USA http://www.csi.ucd.ie/UserFiles/publications/1172146319311.pdf

[Hofmann 2001] Hofmann, T. (2001). Unsupervised Learning by Probabilistic Latent Semantic Analysis. Machine Learning, (pp. 177–196). Kluwer Academic Publishers. http://www.jgaa.info/~th/papers/Hofmann-UAI99.pdf

[Ježek *et al.* 2008] Ježek, K., Fiala, D., & Steinberger J. (2008) Exploration and evaluation of citation networks. *Proceedings ELPUB 2008 Conference on Electronic Publishing - Toronto, Canada.* http://elpub.scix.net/data/works/att/351_elpub2008.content.pdf

[Kiat 2005] Kiat, N. Y. (2005). Citation Parsing Using Maximum Entropy and Repairs. Undergraduate thesis. Department of Computer Science School of Computing. National University of Singapore. http://aye.comp.nus.edu.sg/publications/theses/yongKiatNgThesis.pdf

[Kleinberg 1999] Kleinberg, J. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM* (JACM), 46(5), pp. 604–632. http://www.cs.cornell.edu/Info/People/kleinber/auth.pdf

[Lafferty *et al.* 2001] Lafferty, J., McCallum, A., Pereira, F. (2001). Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. *Proceedings of the Eighteenth International Conference on Machine Learning*, (pp. 282–289). http://www-2.cs.cmu.edu/~mccallum/papers/crf-icml01.ps.gz

[Lausen *et al.* 2005] Lausen, H., Stollberg, M., Hernández, R., Ding, Y., Han, S., Fensel, D. (2005). Semantic Web Portals–State of the Art Survey. *Journal of Knowledge Management*, 9(5), (pp. 40–49). http://www.deri.org/fileadmin/documents/DERI-TR-2004-04-03.pdf

[Newman 2001] Newman, M. E. J. (2001b). Scientific collaboration networks. II. Shortest paths, weighted networks, and centrality. *Physical Review E*, 64(1). http://www.santafe.edu/~mark/papers/016132.pdf

[Newman & Girvan 2004] Newman, M. E. J. & Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical Review E,* 69(2). http://arxiv.org/pdf/cond-mat/0308217

[Nie *et al.* 2007] Nie, L., Davison, B., & Wu, B. (2007). From whence does your authority come? Utilizing community relevance in ranking. *Proc. of the 22nd Conference on Artificial Intelligence (AAAI)*, Vancouver. http://www.cse.lehigh.edu/~brian/pubs/2007/AAAI/community-rank.pdf

[Page *et al.* 1998] Page, L., Brin, S., Motwani, R., Winograd, T. (1998). The PageRank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project. http://citeseer.ist.psu.edu/page98pagerank.html

[Ren *et al.* 2007] Ren, W., Yan, G., Liao, X. (2007). A Simple Probabilistic Algorithm for Detecting Community Structure in Social Networks. http://arxiv.org/pdf/0710.3422

[Studer et al. 1998] Studer, R., Benjamins, V.R., & Fensel, D. (1998) Knowledge engineering: Principles and methods. Data & Knowledge Engineering, 25(1-2), pp. 161–197. Elsevier. http://www.aifb.uni-karlsruhe.de/WBS/Publ/1998/dke98.html

[Sure *et al.* 2005] Sure, Y., Bloehdorn, S., Haase, P., Hartmann, J., & Oberle, D. (2005) The SWRC Ontology-Semantic Web for Research Communities. Lecture Notes in Computer Science. Volume 3808, Springer. http://www.aifb.uni-karlsruhe.de/WBS/ysu/publications/2005_swrc_baosw.pdf

[Šešera *et al.* 2000] Šešera, Ľ., Mičovský, A., & Červeň, J. (2000) Architektúra softvérových systémov. Analytické dátové vzory. Vydavateľstvo STU, Bratislava.

[Tang *et al.* 2007] Jie Tang, Jing Zhang, Duo Zhang, Limin Yao, Chunlin Zhu, Juanzi Li (2007) ArnetMiner: An Expertise Oriented Search System for Web Community. *Frontiers of Computer Science in China.* Volume 2 number 1. pp. 94–105. Springer. http://ftp.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-295/paper01.pdf

[Walker *et al.* 2007] Walker, D., Xie, H., Yan, K.K., & Maslov, S. (2007) Ranking scientific publications using a model of network traffic. *Journal of Statistical Mechanics: Theory and Experiment.* Institute of Physics Publishing. http://arxiv.org/pdf/physics/0612122v1

[White & Smyth 2003] White, S., & Smyth, P. (2003) Algorithms for estimating relative importance in networks. *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 266–275. http://portal.acm.org/citation.cfm?id=956782

[Yolum & Singh 2003] Yolum, P., Singh, M. (2003). Dynamic communities in referral networks. *Web Intelligence and Agent System*, 1(2), pp. 105–116. http://people.engr.ncsu.edu/mpsingh/papers/mas/wias-03-community.pdf

[Zaiane *et al.* 2007] Zaiane, O.R. and Chen, J. and Goebel, R. (2007) DBconnect: mining research community on DBLP data. *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis.* ACM New York, NY, USA, pp. 74–81. http://workshops.socialnetworkanalysis.info/websnakdd2007/papers/submission_1.pdf

[Zhou *et al.* 2002] Zhou, W., Wen, J., Ma, W., Zhang, H. (2002). A Concentric-Circle Model for Community Mining in Graph Structures. *Microsoft Research, Seattle, Technical Report MSR-TR-2002-123*. ftp://ftp.research.microsoft.com/pub/tr/tr-2002-123.pdf

A Technická dokumentácia

Technická dokumentácia obsahuje opis portálového rámca Apache Cocoon, na ktorom je postavený portál, inštalačnú príručku, používateľskú príručku a vývojársku dokumentáciu.

A.1 Súčasti Apache Cocoon

Sitemap je súbor definícií štruktúry adries webového sídla, zvyčajne umiestnený v domovskom adresári príslušného sídla. Súbor obsahuje zoznamy použitých generátorov, transformerov, serializerov, dátovodov, a ich nastavenia. Webové sídlo môže obsahovať viacero máp sídla na rôznej úrovni hierarchie adresárov, čo umožňuje vytvárať zložitejšiu štruktúru sídla.

Generator inicializuje spracovanie dátovodu a vytvára obsah v štruktúrovanom formáte XML[12], sekvenčne prechádza vstupný súbor alebo prúd údajov a počas syntaktickej analýzy generuje udalosti.

Transformer robí mapovanie XML štruktúry na inú XML štruktúru. Transformátor môže byť odvodený alebo môže byť použitý niektorý vstavaný, napríklad ten, ktorý vykonáva transformáciu podľa pravidiel špecifikovaných jazykom XSLT[13]

Serializer vykonáva zápis zo štruktúry XML do špecifikovaného výstupného formátu.

Matcher predstavuje časť spracovania, ktorá hľadá vhodný dátovod definovaný v mape sídla na základe klientom poslanej požiadavky identifikovanej URL adresou z mapy sídla. Môže použiť zástupné znaky (angl. wildcards); alebo regulárny výraz pre skupinu identifikátorov zdroja.

View poskytuje určitý pohľad na dátovod. Umožňuje v dátovode definovať *návestie* (label), a tým vytvoriť odbočku z dátovodu ignorovaním zvyšku pôvodného dátovodu. Následne možno použiť Serializer, ktorý dátovod korektne ukončuje. Každý

[12] Extensible Markup Language http://www.w3.org/XML/

[13] XSL transformations Version 1.0 W3C Recommendation 16. november 1999 http://www.w3.org/TR/xslt

dátovod začínajúci generátorom musí byť ukončený serializerom. Medzi nimi môže byť konečný počet transformátorov.

A.1.1 Spracovanie dokumentov

Proces spracovania XML dokumentov je nasledovný:

1. Pridelenie spracovateľa časťami *Matcher*.
2. Generovanie XML dokumentov (zo statického obsahu, aplikačnej logiky, relačnej databázy, objektov alebo ich ľubovoľnej kombinácie) cez *Generátory*.
3. Transformácia (do iného XML, objektu alebo ľubovoľnej kombinácie) XML dokumentu cez *Transformer*(y)
4. Agregácia XML dokumentov cez *Agregátor*
5. Spodobnenie–*rendering* XML cez *Serializer*

A.1.2 Generátory obsahu dokumentu

Generátory sú rozdelené podľa použitia do troch skupín:

1. štandardný generátor - generátor zo súboru.
2. generátory, ktoré tvoria jadro:
   request generator používa informácie, ktoré poslal klient pri žiadosti.
   JX generator umožňuje prístup k údajom, ktoré odovzdáva FlowScript, poskytuje generickú šablónu so vstavanou podporou jazyka pre transformácie výrazov a naviazanie premenných. Okrem toho umožňuje vkladať do šablóny značky pre: podmienky, cykly, príkazy pre lokalizáciu a formátovanie (locale).
   calendar generator vytvára kalendár.
3. voliteľné generátory
   fragment extractor generator tvorí pár transformátor-generátor, používaný pre vytiahnutie častí XML dokumentu, napríklad vloženého vektorového obrázka SVG a uloženie do grafického formátu PNG alebo JPEG, čo umožňuje posielať obrázky cez samostatný dátovod.
   profile generator vytvára prehľad o čase potrebnom na vykonanie transformácií–*profiling*.
4. odvodené generátory

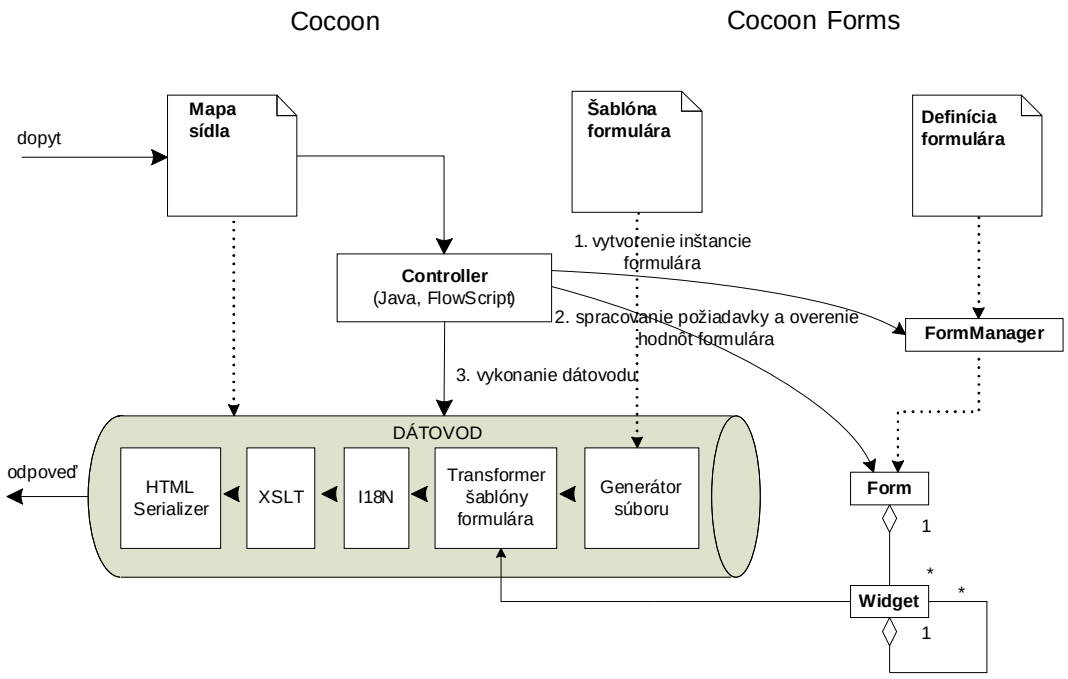
A.1.3 Generátor formulárov

Cocoon Forms umožňujú vytvorenie formulárov podľa šablóny transformáciou XSLT dokumentu vo formáte XML na špecifikovaný HTML dokument s formulárom.

Výhodou takto špecifikovaného formuláru je to, že Cocoon Forms vykonáva overenie hodnôt polí formulára poslaného klientom. Overuje sa formát vstupných údajov a rozsah numerických hodnôt; platnosť vstupov v špecifikovanom tvare, napríklad formát dátumu.

Predbežné overenie môže byť vykonané na strane klienta, napríklad skriptom, aby mohli byť opravené chybné vstupy ešte pred odoslaním formulára. Overenie hodnôt formulára by malo byť vždy vykonané aj na strane servera.

Proces spracovania požiadavky klienta znázorňuje Obrázok 14.



Obrázok 14: Spracovanie požiadavky portálovým skeletom Apache Cocoon

Controller – Cocoon flow skriptovací jazyk *FlowScript* odvodený z jazyka JavaScript umožňuje spracovať požiadavku poslanú klientom a dynamicky vytvárať a pracovať s objektmi jazyka Java.

Generátor súboru – zdrojom vstupu pre generátor je súbor vo formáte XML, napríklad šablóna rozmiestnenia prvkov formulára.
Transformer šablóny formulára – deklarácia a definovanie prvkov formulára, naplnenie prvkov formulára a ich nastavenie.
I18N Transformer – lokalizácia textov portálu a formátovanie čísel podľa zvoleného jazyka
XSLT Transformer – zabezpečuje transformáciu XML dokumentu pravidlami uvedenými v XSLT súbore.

A.2 Inštalačná príručka

Pre inštaláciu produktu je potrebné vykonať tieto operácie v uvedenom poradí.

1. Nainštalovať *Apache Tomcat* (http://tomcat.apache.org/)
2. Nainštalovať *OpenRDF Sesame* (http://openrdf.org/)
3. Nainštalovať *Maven* (http://maven.apache.org/)
4. Skompilovať a nainštalovať zdrojové súbory príkazom `mvn install`
5. Spustiť server v testovacom režime príkazom `mvn jetty:run`

Postup vytvorenia balíka pre Apache Tomcat:

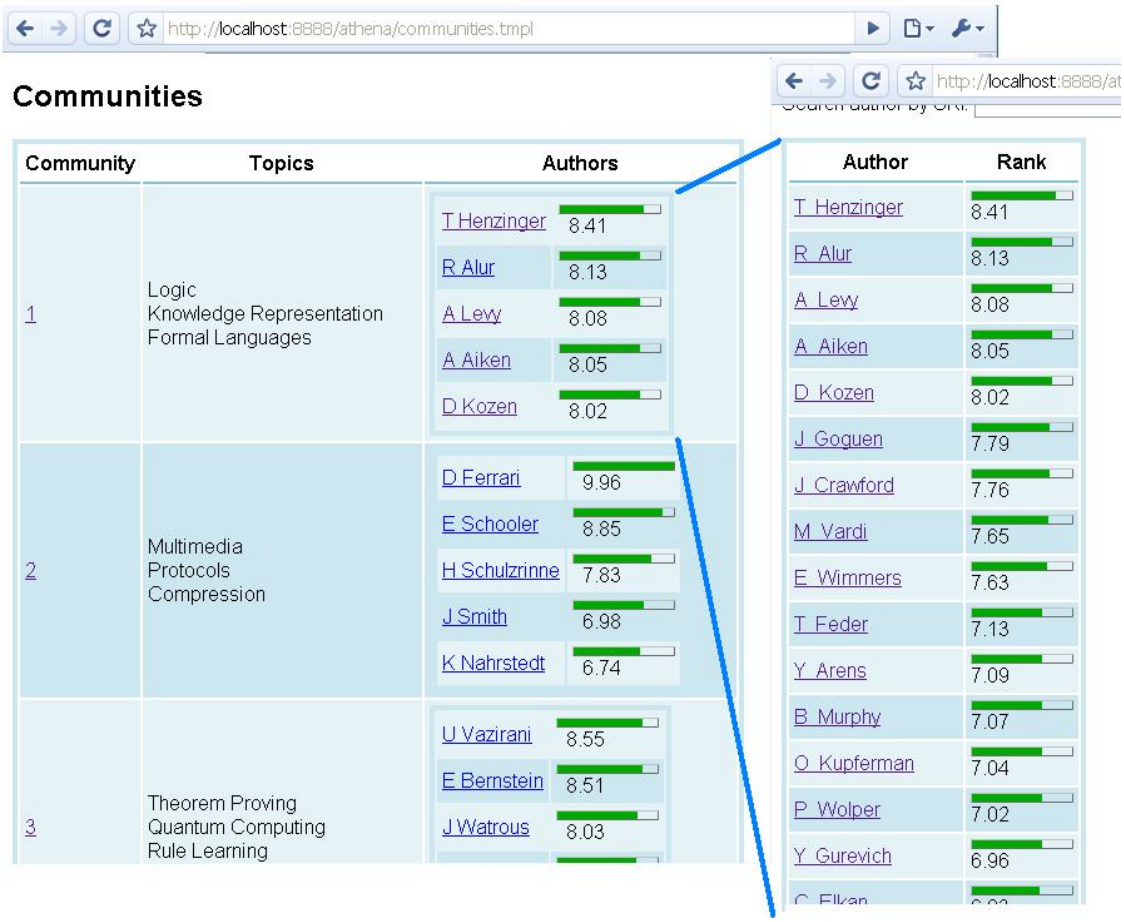
1. V adresári `athena`
   a. vymazať predchádzajúce balíky príkazom `mvn clean`
   b. skompilovať a inštalovať balík príkazom `mvn install`
2. v adresári `athenaWebapp`
   a. vymazať predchádzajúce balíky príkazom `mvn clean`
   b. vytvoriť balík príkazom `mvn package`
3. Vytvorený balík sa nachádza v adresári `athenaWebapp/target` ako súbor `athenaWebapp-1.0-SNAPSHOT.war`
4. Súbor premenovať na `athenaWebapp.war` a nahrať do adresára `webapps`

Ak ontologické úložisko čaká na dopyt na inej adrese ako http://localhost:8080/openrdf-sesame alebo ak je použitá iná ontológia ako `cora`, potom je potrebné v archíve `athenaWebapp.war` editovať súbor `WEB-INF\lib\athena-1.0-SNAPSHOT.jar` a v ňom je potrebné upraviť súbor `athena\src\main\resources\META-INF\cocoon\spring\application-context.xml`, nastaviť hodnoty vlastnosti `server` a `repository`.

A.3 Používateľská príručka

Po inštalácii (v prípade nasadenia do Apache Tomcat) je portál lokálne dostupný na adrese: http://localhost:8080/athenaWebapp/athena/

Na úvodnej obrazovke (obrázok 15) sa nachádza tabuľka komunít autorov spolu s príslušnými témami komunity a piatimi autormi s najvyšším hodnotením v rámci komunity. Aktivovaním odkazu čísla komunity sa zobrazia všetci autori v komunite usporiadaní podľa hodnotenia.



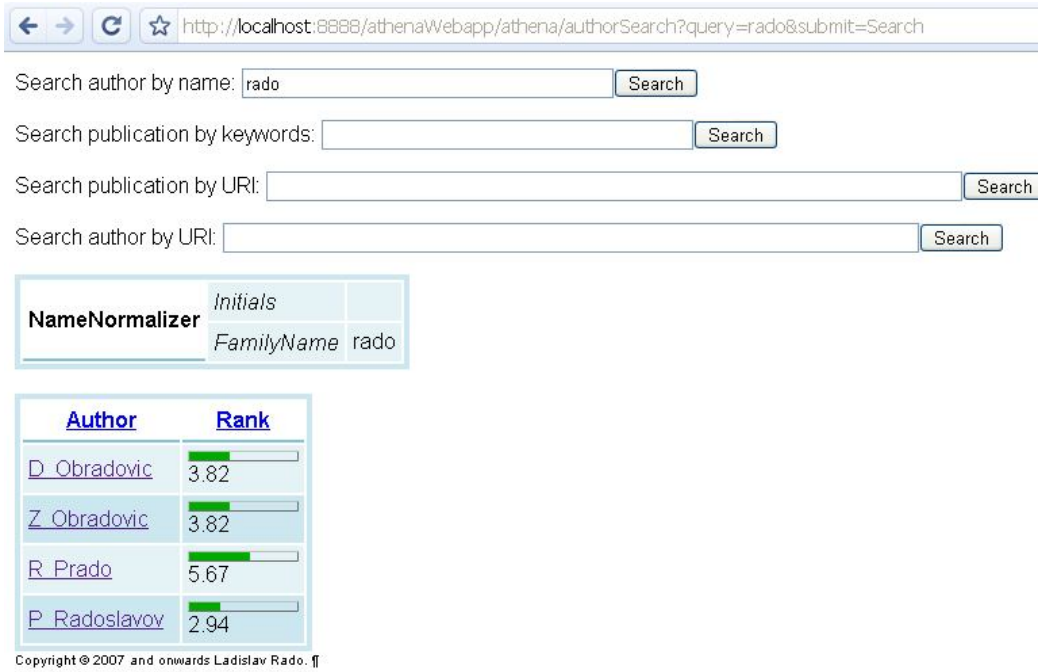
Obrázok 15: Zobrazenie komunít

Aktivovaním odkazu s menom autora sa zobrazia spoluautori a publikácie tohto autora (obrázok 16). Spoluautorov je možné zobraziť graficky, presúvať sa po grafe spoluautorstva. Publikácie autora možno usporiadať zostupne podľa roku alebo hodnotenia.

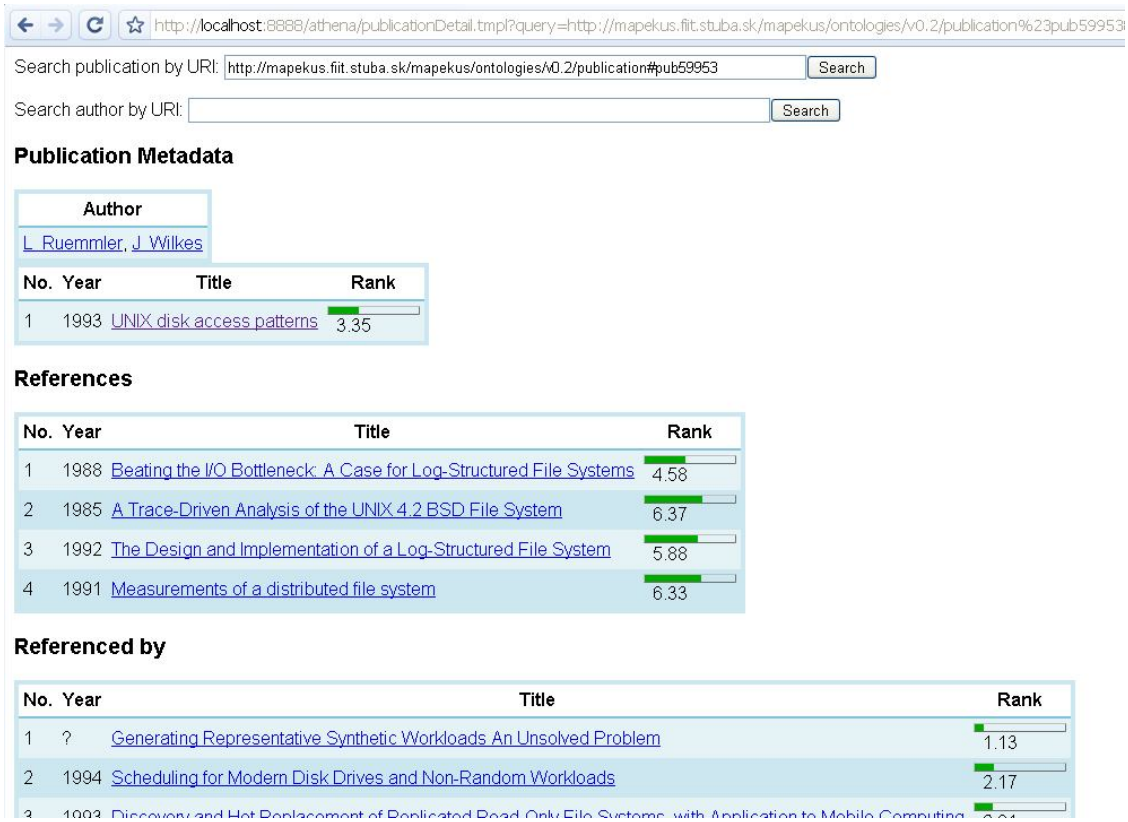
Obrázok 16: Zobrazenie metadát autora

Pre vyhľadanie autora môže použiť formulár pre vyhľadanie autora znázornený na obrázku 17. Pri zadaní prázdneho poľa sa zobrazia všetci autori, čo trvá dlhšie. Používateľ môže okrem toho vyhľadať publikácie podľa kľúčových slov. Prázdne pole pri vyhľadaní publikácií spôsobí zobrazenie všetkých publikácií z ontológie.

Obrázok 18 znázorňuje zobrazenie metadát vybranej publikácie. Možno sa presúvať po grafe citácií nasledovaním odkazu citovaných alebo citujúcich publikácií.



Obrázok 17: Zobrazenie hľadaného autora podľa časti priezviska



Obrázok 18: Zobrazenie metadát publikácie

A.4 Príručka vývojára

Zdrojový kód portálu je organizovaný ako balíky v priestore `src/main/java`, ďalšie zdroje sú organizované v adresári `src/main/resources`

sk.fiit.bean – triedy obsahujú metódy nastavenia a získania hodnôt atribútu triedy.

- Name – reprezentácia mena autora
- Citation – reprezentácia citácie

sk.fiit.config – triedy pre nastavenie parametrov balíka

- ConfigRepository – nastavenie prístupových údajov do ontologického úložiska
- ConfigTopicTree – nastavenie generátora hierachie tém

sk.fiit.generator – odvodené generátory pre Apache Cocoon

- CitationParserGenerator – generátor položiek citácií
- CommmunityGenerator – generátor komunít
- CommunityGraphGenerátor – generátor grafu komunít (nedokončený)
- GraphGenerator.java – generátor grafu spoluautorov vo formáte GraphML
- QueryResultGenerator – generátor výsledkov dopytu do ontológie podľa typu dopytu
- SpellcheckGenerator – generátor opravy preklepov
- TopicTreeGenerator – generátor hierarchie tém

sk.fiit.helper – pomocné moduly

- **CitationParser** – vytvára súbor s črtami pre rozpoznávanie citácií metódou *Conditional Random Fields* (CRF).
    1. Vytvorenie súboru s črtami na základe poskytnutého reťazca referencie.
    2. Načítanie slovníka zo súboru s menami pre vytvorenie modelu charakteristických čŕt.
    3. Spustenie externého procesu, ktorým sa na základe modelu a charakteristických čŕt reťazca referencie priradia značky jednotlivým položkám citácie.
    4. Naplnenie položiek citácie hodnotami výstupu procesu. Úprava mena každého autora na jednotný tvar.
    5. Generovanie výstupu vo formáte XML alebo ako objekt triedy Citation.

- CitationNormalizer - úprava položiek citácie na spoločný tvar

- Diacritics – odstránenie diakritiky
- NameRecognizer – rozpoznávanie tvaru mena

sk.fiit.convert

- AuthorTopicMatrix – načítanie matice autor-téma z ontológie
- ConvertCora – konverzia množiny Cora z textového formátu do relačnej databázy.
- Cora – konverzia z relačnej databázy do ontológie.
- ConvertACM – koverzia

sk.fiit

- PersonalizedTopics– generator personalizovaného stromu
- PLSA – pravdepodobnostná latentná sémantická analýza
- QueryResultParser – spracovanie výsledku dopytu do formátu a
- QueryTypes – typy dopytov
- SesameOntologyRepository – odvodená trieda od SesameHTTPRepository pridáva metódu pre aktualizáciu úložiska Sesame
- TopicTree – reprezentácia hierarchie tém

external – komponenty prevzaté z webu

StringComparer.java – porovnávanie reťazcov Editačnou vzdialenosťou

Spelling – oprava preklepov v mene autora

src/main/resources

`COB-INF` – adresár obsahuje mapu sídla a ďalšie zdroje

`applet` – obsahuje skompilovanú verziu Java appletu

gview.jar – skompilovaný applet pre načítanie a zobrazenie grafu generovaného súborom GraphGenerator.

prefuse.jar – skompilovaná knižnica pre zobrazovanie grafov.

`flow` – obsahuje skripty ovládacej časti portálu – Controller

setuser – obsahuje funkcie pre prihlásenie a odhlásenie používateľa. Nastavuje premennú *session* pre používateľa po jeho prihlásení.

`locale` – adresár obsahuje reťazce pre lokalizáciu textov.

Súbor `locales.xml` – obsahuje zoznam lokalizácií.

`messages_[LC].xml` – texty pre vybraný jazyk určený kódom LC.

`resource` – adresár obsahuje štýly a obrázky.

Adresár `external/` obsahuje hlavný súbor s kaskádovými štýlmi (CSS)

`template` – adresár obsahuje šablóny XML súborov.

- Šablóna pre *zobrazenie detailov autora*: metadáta autora, spolupracovníci autora, témy záujmu autora získané na základe jeho publikácií, a zoznam autorových publikácií
- Šablóna pre *zobrazenie návrhov autorov* po zadaní priezviska autora do formulára pre hľadanie autorov.
- Šablóna pre *zobrazenie rozpoznaných častí citácie* na základe zadania reťazca citácie.
- Šablóna pre *zobrazenie metadát publikácie*: autori publikácie, ostatné metadáta publikácie, zoznam citácií, zoznam publikácií citujúcich publikáciu.

`athena.xslt` – pravidlá pre transformáciu XML do HTML.

- Pravidlá formátujú dokument do požadovaného tvaru ako napríklad generovanie tabuľky metadát, zobrazenie formulárov pre používateľské vstupy.
- Súbor obsahuje šablóny (makrá) pre nahradenie sekvencie znakov inou sekvenciou znakov a pre konverziu URI do formátu ISO-8991-1

`applet.xslt` – vytvorenie stránky, ktorá zobrazuje applet.

`META-INF`

- adresár obsahuje konfiguračné súbory.

Spracovanie dopytov do ontologického úložiska

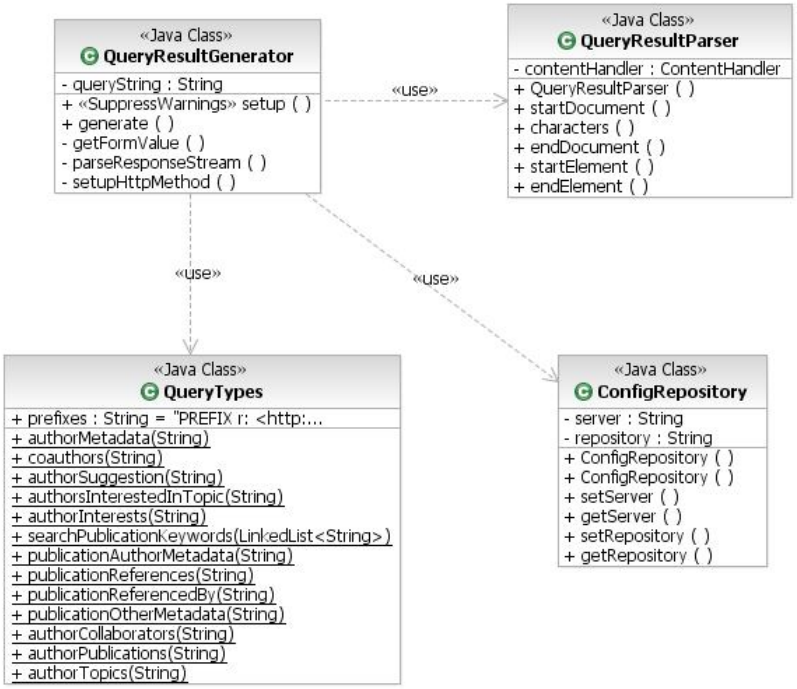


Diagram tried generátora výsledkov dopytu do ontologického úložiska.

«Class» **QueryTypes** – typy dopytov v jazyku SPARQL. Príklad dopytu pre získanie niekoľkých autorít v komunite.

```
public static String communityAuthors(String communityURI, Integer
topK) {
  String limit = "";
  if (topK != null) {
    limit = " LIMIT " + topK;
  }
  return prefixes +
    "SELECT ?authorURI ?givenName ?familyName ?rank WHERE {\n" +
    " ?authorURI publication:hasCommunity <" + communityURI + ">;\n" +
    "   party:givenName ?givenName;" +
    "   party:familyName ?familyName;" +
    "   party:logPageRank ?rank. } ORDER BY DESC(?rank)" + limit;
}
```

«Class» **ConfigRepository** – trieda pre uloženie nastavení do ontologického úložiska.

«Class» **QueryResultParser** – výsledok dopytu jazyka SPARQL vo formáte XML je prevedený na udalosti SAX (Simple API for XML), ktoré Apache Cocoon vnútorne používa.

«Class» **QueryResultGenerator** – generátor výsledného dokumentu podľa typu dopytu.

A.4.2 Rozpoznávanie položiek citácií

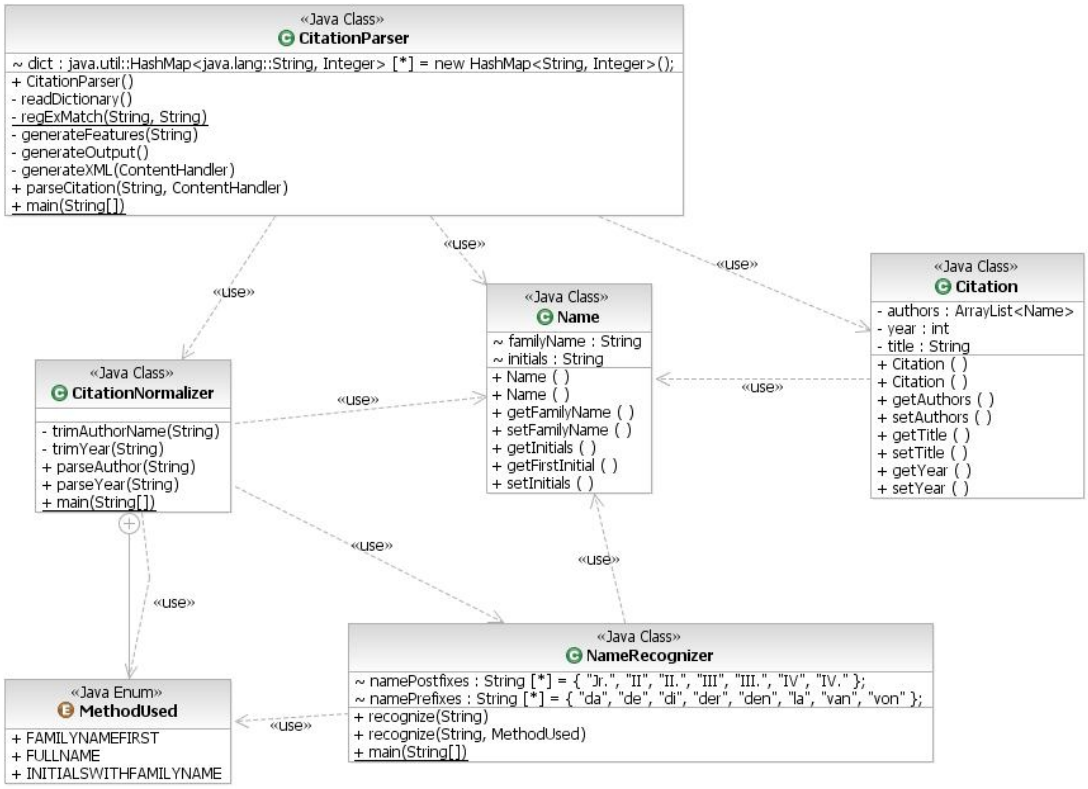


Diagram tried modulu pre rozpoznávanie položiek citácií

«Class» **CitationParser** – rozpoznanie položiek citácií

«Class» **Name** – reprezentácia mena autora

«Class» **Citation** – reprezentácia citácie

«Class» **NameRecognizer** – rozpoznanie tvaru mena, ktorého výsledkom je objekt Name

«Enum» **MethodUsed** – tvar mena autora

A.4.3 Ukážka implementácie generátora

```
public void generate() throws SAXException, ProcessingException {
  String lang = null;
  Integer topK = null;
  while (paramNames.hasMoreElements()) {
    String param = paramNames.nextElement().toString();
    String paramValue = request.getParameter(param);
    if ("lang".equals(param)) {
      lang = paramValue;
    }
    if ("num".equals(param)) {
      topK = Integer.parseInt(paramValue);
    }
  }
  String query = QueryTypes.communities();

  TupleQuery tupleQuery;
  TupleQueryResult result;
  List<? extends BindingSet> resultList;
  ListIterator<? extends BindingSet> iter;
  String communityURI;

  AttributesImpl attributes = new AttributesImpl();
  contentHandler.startDocument();
  contentHandler.startElement("", "communities", "communities", attributes);

  try {
    RepositoryConnection connection = repository.getConnection();
    tupleQuery = connection.prepareTupleQuery(QueryLanguage.SPARQL, query);
    result = tupleQuery.evaluate();
    resultList = Iterations.asList(result);
    iter = resultList.listIterator();

    while (iter.hasNext()) {
      BindingSet bindingSet = iter.next();
      communityURI = bindingSet.getValue("community").stringValue();

      attributes = new AttributesImpl();
      attributes.addAttribute("", "uri", "uri", "", communityURI);

      contentHandler.startElement("", "community", "community", attributes);
      generateCommunityTopics(connection, communityURI, lang);
      generateCommunityAuthors(connection, communityURI, topK);
      contentHandler.endElement("", "community", "community");
    }
  } catch (Exception e) {
    throw new ProcessingException(e);
  }
  contentHandler.endElement("", "communities", "communities");
  contentHandler.endDocument();
}
```

B Štruktúra ontológie publikácií

Ontológia publikácií z projektu MAPEKUS obsahuje triedy: Address, Cluster, Contact_Information, Currency, Event, IndexTerm, Keyword, Language, Party, Person, Project, Publication, Region.

V ontológii sú špecifikované tieto priestory mien (namespaces):

```
xmlns="http://mapekus.fiit.stuba.sk/mapekus/ontologies/v0.2/publication#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:owl2xml="http://www.w3.org/2006/12/owl2-xml#"
xmlns:owl="http://www.w3.org/2002/07/owl#"
xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:party="http://mapekus.fiit.stuba.sk/mapekus/ontologies/v0.2/party#"
xmlns:publication=
"http://mapekus.fiit.stuba.sk/mapekus/ontologies/v0.2/publication#">
```

Triedy bez prefixu sú definované v priestore `publication`.

Trieda Publication sa delí na podtriedy: Article, Book, Journal, Magazine, Newsletter, Paper, Poster, Proceedings, Technical Report, Thesis a Transaction.

Dátové vlastnosti triedy Publication:

- abstract (typu `xsd:string`, funkcionálna),
- day, month, year (typu `xsd:int`, funkcionálna),
- firstPage, lastPage (typu `xsd:string`, funkcionálna),
- pageRank, logPageRank (typu `xsd:double`, funkcionálna),
- source, title, web (typu `xsd:string`, funkcionálna),

Objektové vlastnosti triedy Publication:

- describesProject (multiple Project),
- hasKeyword (multiple Keyword),
- isReferencedBy (multiple Publication),
- isRelatedTo (multiple Event),
- isWrittenBy (multiple party:Person),
- references (multiple Publication),
- similarPublication (multiple Publication).

Rozšírenie ontológie o komunity bez väzobnej entity Membership pridáva:

- Triedu Community s dátovou vlastnosťou *hasTopic* (multiple IndexTerm)
- Objektovú vlastnosť *isMemberOf* (multiple Community) triede party:Person

Rozšírenie ontológie o komunity s väzobnou entitou Membership pridáva:

- Triedu Community s objektovou vlastnosťou *hasCommunityTopic* (multiple CommunityTopic)
- Triedu CommunityTopic s objektovou vlastnosťou *hasTopic* (multiple IndexTerm) a dátovou vlastnosťou *rank* (typu `xsd:double`, funkctonálna)
- Triedu Membership s objektovou vlastnosťou hasCommunity (multiple Community) a dátovou vlastnosťou *rank* (typu `xsd:double`, funkcionálna)
- Objektovú vlastnosť hasMembership (multiple Membership) triede party:Person

C Obsah priloženého média

Obsah adresárovej štruktúry média DVD-R (Digital Versatile Disc):

/	- koreňový adresár média
index.html	- hlavný hypertextový súbor
bibliography/	- adresár obsahuje použitú literatúru
ontologies/	- schéma a inštancie ontológie publikácií
software-external/	- podporný softvér (HTTP server, úložisko,...)
install/	- inštalačné súbory
libraries/	- podporné knižnice a ich zdrojové texty
thesis/	- adresár obsahuje text diplomovej práce
dp.docx	- formát DOC XML (Microsoft Word 2007)
dp.pdf	- formát PDF (Portable Document Format)
workspace/	- adresár obsahuje zdrojový text riešenia
applet/	- Java applet zobrazujúci skupiny v okolí autora
athena/	- webový portál
athenaWebapp/	- balík do servera Apache Tomcat