Wizard-of-Oz-Driven Bootstrapping of a Socially Intelligent Tutoring Strategy

Jozef Tvarožek and Mária Bieliková
Faculty of Informatics and Information Technologies
Slovak University of Technology in Bratislava, Slovakia
jtvarozek@fiit.stuba.sk, bielik@fiit.stuba.sk

Abstract: This research investigates ways to bootstrap a socially intelligent tutoring agent using a Wizard-of-Oz design in which human teachers work as wizards. Students work in an interactive learning environment featuring problem solving and course notes activities, in either individual or group mode. They interact with the agent during off-task dialogs. This paper explores the effort needed on the part of classroom teachers in order to bootstrap an automated agent that can execute near-optimal tutoring actions within the environment. The proposed bootstrapping method can detect peers who will be able to interact well socially, and it can balance individual and social activities for students. Consequently, an automated tutor assisted by classroom teachers, rather than costly expert modeling, can be deployed in interactive learning environments in order to supplement and add to the attention a teacher is able to give students.

Introduction

Time spent on-task and motivation are key factors for effective learning. Therefore, ways for improving motivation in online learning environments are repeatedly studied (Rovai, 2007; Kim, 2009). Children using instructional tools do not use them to their fullest, frequently engaging in off-task behaviors associated with less learning (Baker, 2004). Pedagogical agents used as learning companions have been found to increase student motivation (Kim, 2003; Park, 2007). The pedagogical agent facilitates student's learning in that it can allow the student operate in his/her zone of proximal development. However, deploying sophisticated intelligent agents requires extensive cognitive modeling of the domain which is time consuming even with specialized tools; requiring 200 to 300 hours of development per hour of instruction (Aleven, 2006). Šimko et al. (2010) propose an adaptive learning framework that facilitates active student participation as well as course adaptation and extensible personalization features.

The seminal work of Bloom (1984) on human tutoring provides a strong motivation to deploy such agents in computerized settings. If the same amount of time is used for individual learning with an accomplished tutor, a student's achievements are two standard deviations higher compared to the conventional group instruction – i.e. the average tutored student is above 98% of the students in a conventional classroom. Bloom's study shows that in favorable learning conditions almost all students can reach high levels of achievement. Considering that one-on-one tutoring cannot be implemented on a large scale, we are interested in replicating the success of human tutors in computer-supported learning environments using artificial tutors.

Another promising approach to improve students' motivation is to socialize the learning process. Time spent on-task and the social climate in school are important for good teaching learning (Proctor, 1984). Collaborative learning systems provide software analogies of classroom resources and collaborative activities, such as shared workspaces, synchronous and asynchronous discussions, engaging students in small groups to work together on a common task. Grounded in social cognitive theory (Bandura, 2002), learners are self-organizing and proactive; they ask questions, explain and justify opinions, and reflect upon their knowledge. It is believed that through social interaction and communication, students eliminate misconceptions, gain more in-depth understanding and promote higher-order thinking skills. These benefits are achieved, however, only by active participation and teams that function well (Soller, 2001).

The technology-enhanced learning landscape has been recently influenced by the emergence of social networking. Facebook, the most prominent social networking site, is actively used by more than 500 million users (Zuckerberg, 2010). In common with other social sites, users can create their own personal profiles on

Facebook by providing information that falls into predefined categories, post pictures, participate in discussions, view other peoples' profiles, and communicate with others, link with each other to become "friends," or create and join groups. The potential for educational use of such sites seems tempting. Teachers can create their own profiles, a course page, and use Facebook's functions (e.g. discussion boards, instant messaging) to run the course. However, Facebook does not appear to diminish or eliminate barriers between teachers and students. Students use Facebook for student-to-student exchanges but are less likely to use it in teacher-student interactions (Towner, 2010). Students using an on-line Facebook course tend to engage in passive activities such as viewing others' profiles and reading comments instead of active actions, as, for example, commenting and sending a message. (Teclehaimanot, 2009). In addition, only 66% of the students sampled consider it acceptable to have teachers on Facebook. Acceptance has huge gender differences – 73% of men in the sample consider this acceptable as opposed to only 35% of the women. Issues arise as to what is appropriate for teacher-student interactions. For example, students feel uncomfortable poking or befriending teachers; they also feel uncomfortable when teachers poke or befriend them. Neither Facebook nor other social networking sites have been designed with an educational purpose in mind; educators find it difficult to adjust to them (Cain, 2010). There have also been initiatives aimed at using Web 2.0 principles in web-based learning systems. Straightforward use of Web 2.0 tools creates problems with respect to how participation in the Web 2.0 activities translates into learning outcomes (Dohn, 2009).

In our research, we investigate ways for deploying intelligent pedagogical agents without the need for extensive domain modeling. We propose a novel learning framework in which a socially intelligent agent (tutor) guides students through appropriate instructional activities. The tutoring strategy is improved continuously using a socially augmented reinforcement learning method. In addition to the ordinary exploratory part of reinforcement learning, human wizards provide improved guidance in the state and action spaces. In the evaluation in a simulated learning environment, we observe that a reasonable number of human actions are sufficient to bootstrap the tutoring strategy that is followed by the wizard.

The Socially Intelligent Learning Environment

In this section we present our prototype learning environment. The prototype system is an interactive web-based environment that helps students learn using a variety of learning opportunities facilitated by a socially intelligent tutoring agent. It features pseudo-tutor assessments with free-text answering. Questions for assessments and exercises are generated by a task generator discouraging cheating and surface learning. It is an attempt to build an integrated environment for: (1) assessments and instruction as they occur in regular classrooms; and (2) home study with self exercises. The available learning opportunities are augmented with social features. In order to enable socially intelligent instruction, the tutoring strategy operates on top of socially augmented components that constitute the learning environment: (1) problem solving, (2) course notes, and (3) off-task social dialogs. The components are, then, integrated by the tutoring strategy. A conceptual diagram of a student working in the proposed learning environment is provided in Figure 1.

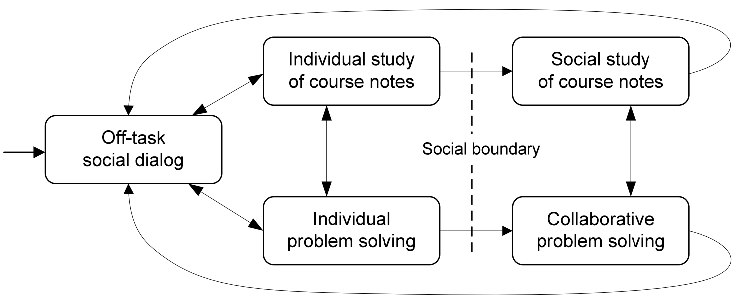

Figure 1: Diagram of various activities that are facilitated by the tutoring strategy.

Using this approach, we attempt to model a spectrum of learning activities – individual and group, and active (problem solving) and passive (reading course notes) – while the tutoring strategy integrates them in a learning experience, and effectively, decides on the pedagogy. That is, we do not rule out any single learning theory, but we also do not choose a single one. No single learning theory works for everyone, and the reinforcement learning approach to modeling a tutoring strategy can (over time) decide on a suitable tutoring strategy (a selection of activities) for each student.

The approach is more suitable for structured domains (e.g. mathematics, programming) as they can better take advantage of the structured problem solving facility. In order to conduct classroom experiments, the system also contains generic social features such as messaging, friend management, invitations, and updating (the individual) profile. The system is both an individual learning environment and a collaborative learning system, and hence is designed as a client-server application. The implementation enables stateful low latency interaction required for synchronous activities.

Student Environment

The student begins with a dialog with the tutoring agent (Figure 2a) in which his immediate goals are determined, and combined with long-term goals and tendencies; the tutor recommends a learning activity to pursue. Either study of course notes or problem solving is selected. For group mode learning activities, students can assemble collaborative groups from their available friends; anonymous introductions can be facilitated by the tutoring agent. Interactions between users are restricted to friends or tutor-recommended students only; hence a student does not come into contact with any entity to which his/her relationship cannot be predicted.

(a) (b)

Figure 2: Prototype system's screens: welcome dialog with the tutor (a), friends management facility (b).

A student can work both actively (problem solving) or passively (course notes), and individually or in collaboration with others. During problem solving, the student is presented with a subtask to solve in a linear manner. When an answer is submitted, its correctness is judged, and a new subtask for the student to solve is displayed allowing the student to scroll all the way back to the beginning and see the course of action he/she took. A free text prompt is used to answer subtasks.

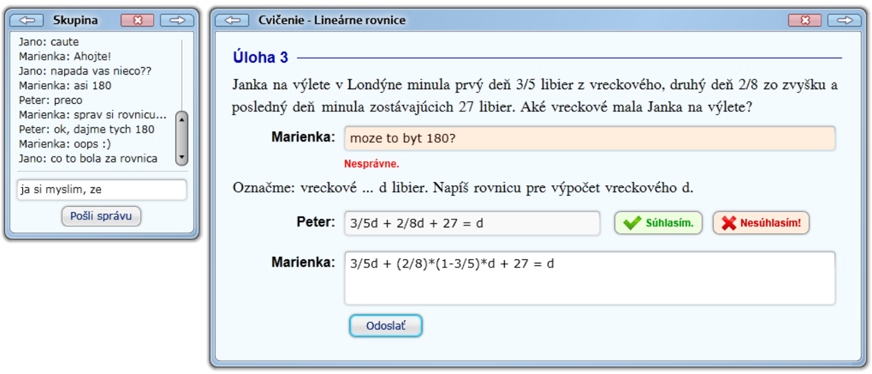

Figure 3: Collaborative problem solving – free-text task answer with voting mechanism.

In the social mode, students can work on problems in groups. Additional interactions such as synchronous messaging within the group and a voting facility are utilized (Figure 3). We do not describe the course notes facility in this paper. Other generic social features include sending messages, friend management (Figure 2b), invitations, and updating the individual's user profile. When a student assembles a study group consisting of friends, he/she is provided with the live status of the friends (contacts) so that the group will be comprised of active users.

Wizard Interface

On the other side of the barricade is the user interface for human wizards. The human wizard must analyze student behavior and take appropriate action. Obviously, a wizard (or two) cannot be provided for each student. The human wizard's interface (Figure 4) is intended to streamline wizard activity to service multiple students simultaneously.

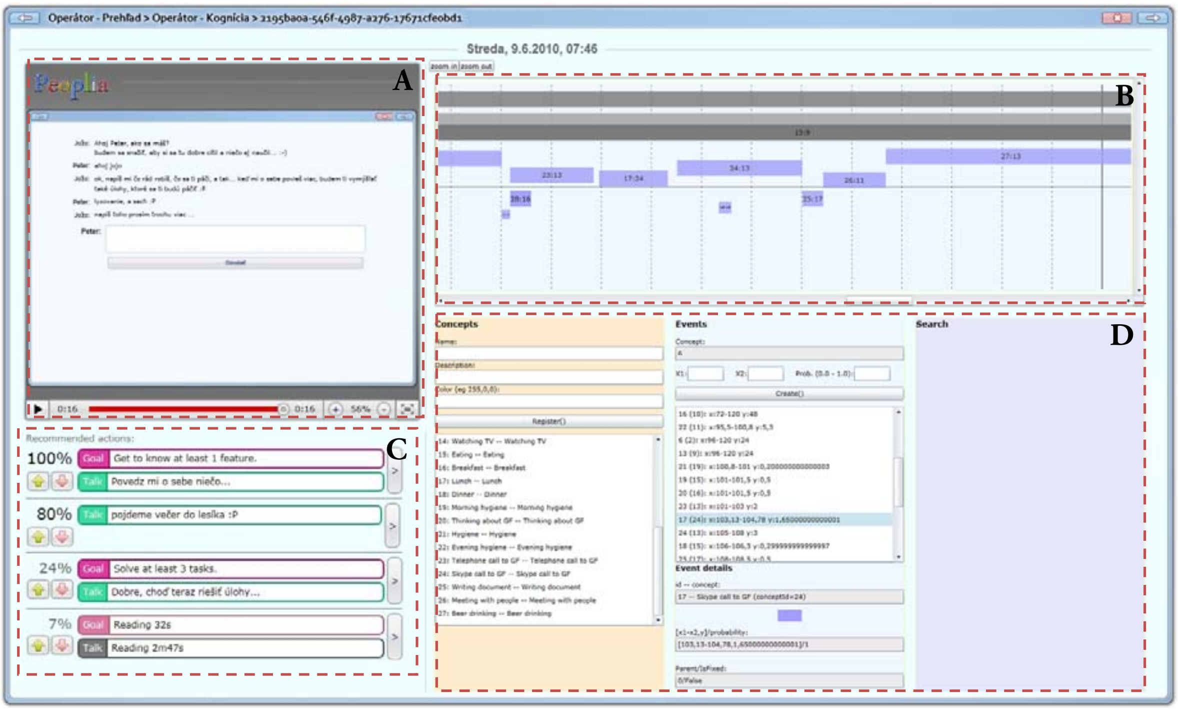

Figure 4: Experimental wizard interface, featuring live/replay-able tracking of student's user interface (A), student's events timeline (B), recommended actions to execute (C), and tools for editing the timeline (D).

During problem solving, the wizard is expected to judge student answers in a timely manner. Judging an answer requires the wizard to manually select a suitable category from a set of pre-defined categories. In an earlier experiment (Tvarožek, 2008), we observed that a single wizard is able to judge student answers for a class of 20 to 30 students without any noticeable lag.

The wizard can observe student's state of client environment, and is capable of executing user interface actions within the client environment. Past student actions in the session can be replayed (Figure 4A) from the very beginning watching the succession of events that led to the current environment state. The socially intelligent tutor (server) analyzes the client's environment state continuously using feature detectors. When a state is recognized recommended action for the human wizard to execute is offered (Figure 4C).

Course Design

Teachers can use the prototype system to design blended courses. In a blended course, students work on exercises during classroom time, while human teachers provide them with immediate assistance and/or timely instruction. After class, students can resume unfinished work and/or further strengthen their understanding at home using the identical online learning environment.

The course curriculum involves planning for activities within individual lessons, as well as activities between the lessons. The teacher plans for timed activities (e.g. assessment, collaborative activity) within a lesson. During the actual lesson, the timing is enforced by the system. Hence, the teacher is in complete control of student use of learning system time and means in class. Outside-lesson activities are also planned by the teacher beforehand. These activities are pursued freely by the students. They can strengthen previous learning experiences, as well as prepare students for the next lesson. The core idea is that teacher plan both in-lesson and outside-lesson activities ahead of time. Consequently, the reinforcement learning-based intelligent agent can base its behavior (recommendations to students) on the teacher-planned curricula. This lets the agent guide students to learning experiences that are relevant for their imminent school needs, providing yet another bit of motivation for students to actually listen to the artificial tutor.

Evaluation Studies

In previous evaluation studies (Tvarožek, 2010) we observed that without any instructions given before the treatment some 56% of students engaged in the dialog with the tutor, revealing on average 1.56 (st.dev 1.75) features (e.g. *I like swimming*, *watching TV*) about themselves. Moreover, students that engaged in a conversation with the tutoring agent exhibited higher learning gains. The not engaged group showed relatively low learning gain 3.7% vs. 12.3% exhibited by the engaged group.

In summary, students that engaged in social off-task dialogues with the tutor were more effective in solving problems correctly (57% vs. 37%), and liked the system more (4.22 vs. 2.86 points out of 5 in a questionnaire), suggesting that learning environments may produce higher learning gains by "being friends" with the students, providing them with socially relevant motivation. A single human teacher cannot provide for each student in the class individually during instruction. The proposed apporach to a socially intelligent tutor is designed to fill this gap when students use computer-supported learning environments.

Bootstrapping the Socially Intelligent Tutoring Strategy

The components of our learning environment can operate in individual and social modes and provide diverse opportunities for instruction. The role of the socially intelligent tutor, then, is to identify an appropriate learning activity for each student at a given time. In other words, the tutor recognizes the student's state and performs an action that guides him/her to the appropriate learning activity. In this section, we propose a method that integrates the individual components into a tutoring strategy, i.e. a policy that can select an appropriate action for the tutor to take in the learning environment.

Method Overview

The proposed method for bootstrapping a socially intelligent tutoring strategy is based on reinforcement learning that is modified to be able to optimize the policy for a single student efficiently as well as when rewards for many students need to be considered, i.e. the method is suitable for social learning environments in which a large number of students participate. In this section, we: (1) describe the state-action view of how the learning environment can be interpreted as a reinforcement learning problem; (2) present the reinforcement learning method for a solitary student; and (3) augment the method with a social graph in order to handle large number of students efficiently

Reinforcement learning is a machine learning approach that induces the optimal policy for an agent to follow in a stochastic environment while, at the same time maximizing cumulative rewards (Sutton, 1998). The agent repeats the following steps indefinitely. At each step:

1. the agent observes the environment recognizing the current state;
2. selects an action to perform;
3. a reward (scalar in R) may be received from the environment; and
4. the environment changes according to the action executed.

The idea is to observe which actions (solving a problem, reading a course note, interacting with a peer, interacting with the tutor) produce positive rewards, and progressively calculate better estimates of action utility values in each state. Ultimately, the policy should be able to select "good" actions in each possible state. Positive rewards are received when a problem is solved successfully or the student demonstrates positive emotional dialog utterances, etc. Similarly negative rewards are received when an inappropriate behavior is observed. Depending on what we want to achieve.

How do we translate our learning environment to this formalism? We cannot possibly hope to observe what students actually do at their computer desks; we can only observe the actions that students have performed within the learning environment. Sets of features of the current situation in the learning environment constitute states which, in turn, constitute the total state space for an individual student in our learning environment. Actions – user interface actions, learning activity actions, or social actions – that are executed in the learning environment are based on action templates that were originally performed by human wizards, i.e. there is no "chaotic" exploration of the state and action spaces. Finally, at times when an activity produces positive/negative results, multidimensional rewards are received (task vs. non-task, emotional vs. social). To find a globally optimal tutoring strategy, the computer tutor cannot follow two independent goals at once but needs to assign weights to the respective reward dimensions. For example, an individual learning-only strategy, a group learning-only strategy, an emotional-only, a social-only, or a weighted combination of any of these strategies can be followed, but in the end, a single scalar reward in R must be received after executing an action.

In addition to traditional exploitation (selecting the best possible action at a state) and exploration (selecting a random action) we extend the reinforcement learning method with human wizard interventions (Figure 5). Wizard actions are welcome at any time, but the assumption is that because the wizard is a real person his/her actions are expensive. As a result, we are interested in minimizing the number of human wizard actions required to learn the optimal policy.

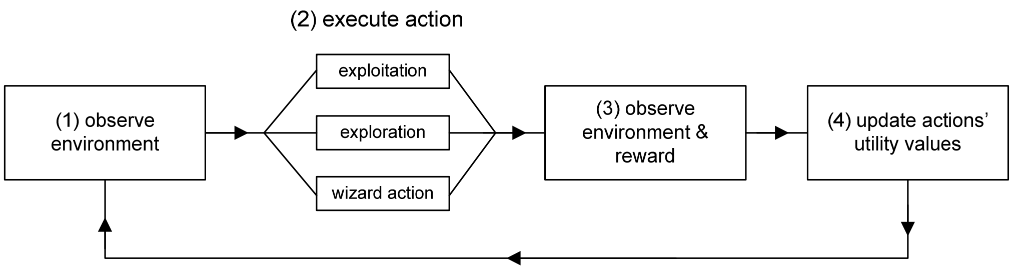

Figure 5: Reinforcement learning approach enhanced with Wizard-of-Oz action selection.

The generic reinforcement learning approach can be used for optimizing the tutoring strategy for a single (solitary) student in the environment. The approach, however, is not applicable to learning optimal policy in a large population of students. We have proposed a modified algorithm that in the case of a social scenario (when multiple students within the learning environment are be considered) aggregates the individual states of other students into a combined shared state, thus reducing the state space to a manageable size. Hence, the proposed method can run efficiently even when a large population of students is serviced.

The environment's state and action spaces are practically infinite. In a face-to-face tutoring session, the human tutor may employ any enlightened tutoring approach; obviously the computer tutor cannot tractably enumerate and explore all the options available in the computer setting. Therefore, guidance by human wizards in exploring applicable actions is indispensable for the computer tutor to obtain knowledge of the relevant states.

Evaluation

We evaluated the feasibility of the proposed approach in a series of simulated scenarios: solitary student scenario, social-only scenario and mixed scenario. The scenarios simulate a wizard's decisions using a hidden model of the environment allowing us to explore the way in which the number of wizard actions affects bootstrapping efficiency.

In each scenario the activities for a student or a population of students are optimized. The activities are organized in a timeline containing *N* action slots. In each slot, the student can perform one of two actions *RELAXING* or *LEARNING*. In social and mixed scenarios the students can perform an additional action *SOCIAL-LEARNING* during which a peer student from the population needs to be selected and the action is performed by both students in the same time slot.

The actual learning of a student is modeled using cyclical efficiency, that is, each person has a single peak mental capacity at a time of day; possibly different across persons. Initially, the student has zero energy, and has learned nothing. When the *RELAXING* action is performed, the student gains two energy units. Energy is needed for effective learning although learning has variable efficiency depending on the time step executed. The *LEARNING* action consumes one unit of energy. When enough energy is available, the student will learn an amount equal to the current value of learning efficiency after consuming one unit of energy. When not enough energy is available, the *LEARNING* action uses the remaining energy and the student learns the amount of learning efficiency proportional to the energy spent.

In this scenario, the individual learning of student *u* in step *i* has cyclical efficiency:

$$individualEfficiency_{i,u} = \frac{1}{(seed + i)_{\mathrm{mod}\,cycle}}$$

Rewards in the reinforcement learning method are given only for *LEARNING* actions and are equal to the amount of learning that occurred. In a way, the cyclical nature of rewards reflects the student's daily routine. The tutoring policy selects actions for each successive step sequentially meaning that the tutor recommends that the student perform the selected action. In a real world setting, the student is free to choose which action he/she will perform; in this scenario, we assume the student executes the requested (tutor-recommended) action.

In social scenarios, the learning of *M* students is optimized simultaneously. Each student has a timeline consisting of *N* action slots. The *RELAXING* action is used to acquire two energy units as was the case in the previous scenario. Similarly, the *SOCIAL-LEARNING* action requires one unit of energy to be effective; however, this (social) action must be undertaken with another student. Because a joint effort is required, the given *SOCIAL-LEARNING* action is executed in both the student's environment and in a peer's environment, that is, in another student's timeline. When no peer is selected for a learning action, its efficiency drops to zero. Each student has a predefined value ($hobby_i$) of interest in a hobby (for simplicity of the scenario, the hobby is the same across students); the closer the values of the two students' interests are, the better the fit for the social learning activity. As a result, when an unsuitable peer is selected, learning efficiency is lowered proportionally to the mismatch.

The individual learning efficiency is further affected by the social efficiency as follows:

$$socialEfficiency_{u,w} = (1 - |hobby_u - hobby_v|)^2$$
$$learningEfficiency_{i,u,w} = \left(\min_{s \in \{u,w\}} individualEfficiency_{i,s}\right) \cdot socialEfficiency_{u,w}$$

As previously, scalar (in R) rewards are awarded to students for learning activities depending on efficiency (learning that occurred); in other words, a student can learn at any time with any peer, but may end up learning nothing (or very little) if the circumstances including being relaxed, or collaborating with a suitable peer are not sufficiently satisfied.

Using this model, optimal action can be selected at each time for each student. The model of individual learning efficiency intuitively follows student's daily routine, and the fact that different students may reach their peak mental efficiency at different times of day. Similarly, the social efficiency model intuitively reproduces a possible indirect relation of group efficiency to individual member's efficiency via hobbies. In practice, however, the individual efficiency is estimated based on student's past interactions, and (should it be found too unfitting) the social efficiency model can be altered once significant data becomes available. The bootstrapping method is not bound to any particular model, and hence we believe that the improvements translate to other (possibly implicit) models as well. The following experiments are simulations of Wizard-of-Oz design. That is, the model is used to simulate human actions at a given participation rate. There are no actual human operating in the experiment runs.

We only present results for the more complex social scenarios. The experiment was run for *N*=20, *cycle*=3, *M*=10; *seed* was set randomly for each student. For each wizard participation rate, the experiment was run 100 times, and the results averaged. Within the reinforcement learning method we used typical values for learning rate 0.1, and discount factor 0.8. The learning speed for selected wizard participation rates is shown in Figure 6. With at least some (5%) wizard participation, roughly 85% of the expected maximum policy can be learned. Interestingly, participation of 10% is sufficient to induce a good tutoring strategy rapidly in the first five epochs. Note that in a single epoch the actions for all of the students are selected, i.e. actually M times more action selections (and reinforcement learning updates) are performed compared to when the same number of epochs were used in the individual scenario.

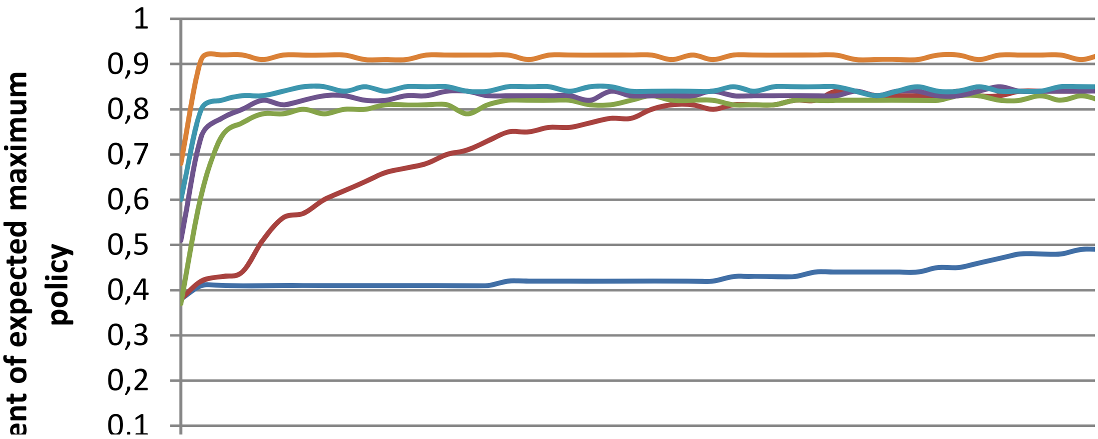

Figure 6: Learning the social-only policy with different wizard participation rates.

In the mixed scenario (where students can perform both individual and social actions, influencing each other), the interaction between individual and social actions caused the bootstrapping method to take longer. Wizard participation of 10% is sufficient to attain 85% to 90% of the theoretical maximum expected mixed (social and individual) policy. Higher wizard participation does not affect the quality of the policy identified but only improves the learning speed, and can produce a good policy very early in the bootstrapping process. The learning speed for selected wizard participation rates is shown in Figure 7.

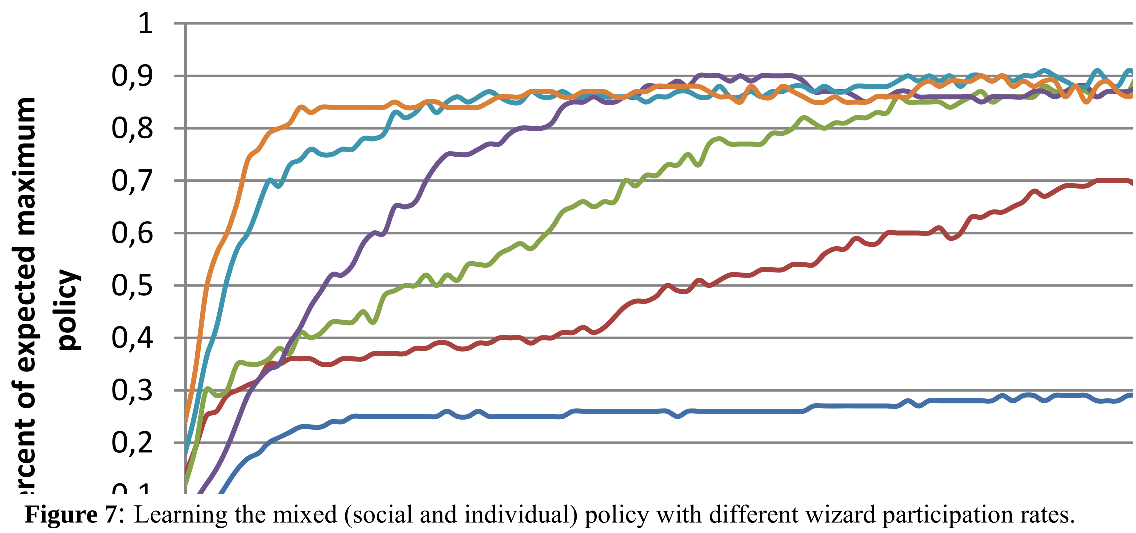

Figure 7: Learning the mixed (social and individual) policy with different wizard participation rates.

The interaction between individual and social actions caused the bootstrapping method to take longer. Wizard participation of 10% is sufficient to attain 85% to 90% of the theoretical maximum expected mixed (social and individual) policy. Higher wizard participation does not affect the quality of the policy identified but only improves the learning speed, and can produce a good policy very early in the bootstrapping process.

The socially augmented approach used for social scenarios is able to detect peers who will be appropriate for effective social interactions. When individual and social actions need to be balanced, the method may take longer, and greater wizard participation will be needed to attain the theoretical maximum for the expected policy. Wizard participation of between 10% and 20% appears to be sufficient to bootstrap policy that will be followed by human wizards. In other words, the simulations suggest that a socially intelligent tutoring strategy that allows the computer tutor to manage students in the proposed learning environment can be bootstrapped with a reasonable amount of human participation.

Conclusions

This paper investigated a Wizard-of-Oz-driven reinforcement learning approach to bootstrap a socially intelligent tutoring strategy that operates on top of typical components of learning environments. An automated computer tutor that follows the strategy can select appropriate learning activities for students. The method is designed to be able to balance individual and social, and cognitive (on-task) and affective (off-task) activities. Combined with the off-task dialog facility guided by human wizards, the proposed bootstrapping method is designed to provide novel interaction patterns in order to maintain their motivation and increase the time students invest in study. Human wizards are costly compared to fully automatized tools. The evaluation in simulated scenarios suggests that as much as 10% of wizard participation in making decisions is sufficient to learn an 80% optimal policy i.e. the tutoring strategy followed by the human wizards. This suggests that the (typically demanding) domain modeling by artificial intelligence engineers can be, to an extent, replaced by less qualified teachers working as human wizards. In the future, we aim to conduct a follow up study in real world setting with the human teachers.

The proposed socially intelligent tutoring strategy is designed to pick suitable learning activities for each student depending on the current state of the student's ability. Using this approach, different pedagogical styles and strategies can be varied to fit the needs of the learner. In the current prototype environment, either (1) individual or group work; or (2) problem solving or working on course notes can be used. Various pedagogical strategies not often used in regular classrooms can be employed on top of these resources, e.g. collaborative learning, observational learning, peer learning, and others. All too often, teachers are currently unable to these pedagogical strategies in classrooms because of their mixed effects on different students. Using the proposed approach however, the learning experience can be personalized differently for each student.

Acknowledgment

This work was partially supported by the grants VG1/0508/09, KEGA 345-032STU-4/2010 and it is the partial result of the Research & Development Operational Program for the project Research of methods for acquisition, analysis and personalized conveying of information and knowledge, ITMS 26240220039, co-funded by the ERDF.

References

Aleven, V., McLaren, B. M., Sewall, J. & Koedinger, K. R. (2006). The Cognitive Tutor Authoring Tools (CTAT): Preliminary Evaluation of Efficiency Gains. In *Proceedings of the 8th International Conference on Intelligent Tutoring Systems* (pp. 61–70), Taiwan, Springer.

Baker, R.S., A.T. Corbett, K.R. Koedinger, & A.Z. Wagner. (2004). Off-Task Behavior in the Cognitive Tutor Classroom: When Students "Game The System." In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 383–390), ACM, 2004.

Bandura, A. (2002). Social cognitive theory of mass communications. In J. Bryant, & D. Zillman (Eds.). *Media effects: Advances in theory and research* (2nd ed., pp. 121-153). Hillsdale, NJ: Erlbaum.

Bloom, B.S. (1984). The 2 Sigma Problem: The Search for Methods of Group Instruction as Effective as One-to-One Tutoring. *Educational Researcher*, 13(6), pp. 4–16.

Cain, C., Seals, C. & Nyagwencha, J. (2010). Social Networking Teaching Tools: A Computer Supported Collaborative Interactive Learning Social Networking Environment for K-12. In: *Proc. of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2010* (pp. 1612–1617), AACE.

Dohn, N. (2009). Web 2.0: Inherent tensions and evident challenges for education. *International Journal of Computer-Supported Collaborative Learning*, 4(3), pp. 343–363.

Kim, K.J. (2009). Motivational Challenges of Adult Learners in Self-Directed e-Learning. *Journal of Interactive Learning Research*, 20(3), 317-335. Chesapeake, VA: AACE.

Kim, Y. (2003). Things that Make Agent as Learning Companion Effective. In A. Rossett (Ed.), *Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2003* (pp. 1659-1666). Chesapeake, VA: AACE.

Park, S., Lim, J., McBride, R., McFerrin, K. & Kim, K. (2007). Designing Effective On-line learning Environments Using Emerging Educational Technologies. In *Proceedings of Society for Information Technology & Teacher Education International Conference 2007* (pp. 464-471). Chesapeake, VA: AACE.

Proctor, C. (1984). Teacher expectations: A model for school improvement. *The Elementary School Journal*, pp. 469–481.

Rovai, A., Ponton, M., Wighting, M. & Baker, J. (2007). A Comparative Analysis of Student Motivation in Traditional Classroom and E-Learning Courses. *International Journal on E-Learning*, 6(3), 413-432. Chesapeake, VA: AACE. 20022.

Soller, A. (2001). Supporting social interaction in an intelligent collaborative learning system. *International Journal of Artificial Intelligence in Education* 12, pp. 54–77.

Sutton, R.S. & Barto, A.G. (1998). *Reinforcement Learning: An Introduction*. MIT Press, Cambridge.

Šimko, M., Barla, M. & Bieliková, M. (2010). ALEF: A Framework for Adaptive Web-Based Learning 2.0. In *Proceedings of IFIP Advances in Information and Communication Technology*, Vol. 324/2010, (pp. 367–378), Springer.

Teclehaimanot, B. & Hickman, T. (2009). Student-Teacher Interaction on Facebook: What Students Find Appropriate. In: *Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2009* (pp. 3181–3190), AACE.

Towner, T.L. & Muñoz, L.C. (2010). Let's 'Face' It: Facebook as an Educational Tool for College Students. In: *Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2010* (pp. 1953–1958), AACE.

Tvarožek, J., Kravčík, M. & Bieliková, M. (2008). Towards Computerized Adaptive Assessment Based on Structured Tasks. In: *Proceedings of Adaptive Hypermedia and Adaptive Web-Based Systems (AH 2008)*, LNCS, Vol. 5149, Hannover, (pp. 224–234), Springer.

Tvarožek, J. & Bieliková, M. (2010). Feasibility of a Socially Intelligent Tutor. In: *Proceedings of Intelligent Tutoring Systems (ITS 2010)*, LNCS, Vol. 6095, Pittsburgh, USA, (pp.423–425), Springer.

Tvarožek, J. & Bieliková, M. (2010). Enhancing Learning with Off-Task Social Dialogues. In: *Proceedings of European Conference of Technology-Enhanced Learning (EC-TEL 2010)*, LNCS, Vol. 6383, Barcelona, Spain, (pp. 445–450), Springer.

Zuckerberg, M. (2010). *500 Million Stories*. July 21, 2010.
URL: http://blog.facebook.com/blog.php?post=409753352130 (accessed December 16, 2010).