

SOCIAL WEB TECHNOLOGIES TO ENHANCE TEACHING AND LEARNING

Vlad POSEA¹, Traian REBEDEA², Costin CHIRU³, Ștefan TRĂUȘAN-MATU⁴

Lucrarea descrie un experiment complex de e-learning la care au luat parte peste 700 de studenți care au participat la cursul de Interacțiune Om-Calculator în ultimii 4 ani. Experimentul a constatat în utilizarea tehnologiilor caracteristice web-ului social, precum blogurile și conferințele chat cu scopul de a implica studenții într-o învățare colaborativă. Sunt prezentate scenariile de învățare, problemele întâlnite în implementarea lor, precum și instrumentele dezvoltate pentru rezolvarea acestora și pentru a asista tutorii la evaluarea activității studenților. De asemenea, sunt acoperite principalele rezultate ale experimentului și ale utilizării instrumentelor de analiză. În plus, vrem să prezentăm beneficiile utilizării unui astfel de scenariu ca supliment la paradigma clasică de predare și învățare, pentru comunitatea de învățare formată din studenții cursului.

The paper describes a complex e-learning experiment that involved over 700 students that attended the Human-Computer Interaction course during the last 4 years. The experiment consisted in using social web technologies like blogs and chat conferences to engage students in collaborative learning. The paper presents the learning scenarios, the encountered problems and the tools developed for solving these problems and assisting tutors in evaluating the activity of the students. The results of the experiment and of using the analysis tools are also covered. Moreover, we show the benefits of using such a scenario for the learning community formed by the students that took this course to supplement the classical teaching and learning paradigm.

Keywords: Social Web, E-learning, CSCL, Blog, Chat, Learning Communities

1. Introduction

The Social Web [1] is the term used to describe the current level of interaction between people using the Internet. It represents a complex evolution

¹ Ph.D. student, Computer Science and Engineering Department, University POLITEHNICA of Bucharest, Romania, e-mail: vlad.posea@cs.pub.ro

² Ph.D. student, Computer Science and Engineering Department, University POLITEHNICA of Bucharest, Romania, e-mail: traian.rebedea@cs.pub.ro

³ Ph.D. student, Computer Science and Engineering Department, University POLITEHNICA of Bucharest, Romania, e-mail: costin.chiru@cs.pub.ro

⁴ Prof., Corresponding member of the Academy of Romanian Scientist, Computer Science and Engineering Department, University POLITEHNICA of Bucharest and Romanian Academy Research Institute for Artificial Intelligence, Romania, e-mail: stefan.trausan@cs.pub.ro

from the first steps of the web in the 90s that were represented by simple web pages containing only text, images and hyperlinks. The Social Web (or Web 2.0) provides complex applications that allow people to create and share content, to discuss, and group themselves into communities.

The concept of Web 2.0 is also related to the socio-cultural ideas introduced by Vygotsky [2], stating that knowledge is socially constructed. Moreover, it can be very well put to use in the educational processes through Computer Supported Collaborative Learning (CSCL, [3]), more precisely as an instrument for putting together knowledge building communities.

This paper presents a series of CSCL experiments that took place at the “Politehnica” University of Bucharest at a course on Human-Computer Interaction for the Computer Science and Engineering students. Web technologies and tools were in the focus of this course for many years. Here was probably the first place in Romania where XML and the associated languages, standards, technologies and tools were taught, even from year 2000. In the assignments and laboratory activities associated to the course, the students had to design and implement advanced web applications. In 2005, after a collaboration with Drexel University in Philadelphia, US (see <http://vmt.mathforum.org/vmt/>), a special emphasis was put on teaching the collaborative and social based tools for the web. One of the main lessons learned was that to efficiently learn something, you have to engage a community of peers to inter-animate when they learn together collaboratively using social web tools like instant messaging [4, 5]. Therefore, we decided to teach social web technologies like chats, forums, blogs, social networks, by using exactly this kind of tools. As a consequence, for 4 years in a row (2005-2008), several assignments were proposed to the students in order to establish a medium sized knowledge building community using Social Web instruments. In the first two years students had up to three assignments to be performed in small groups using instant messenger chat conferencing [4]. In the last two years, the range of experiments was extended to the use of blogs and social networking. In addition, the students used discussion forums that are available in the Moodle LMS (<http://www.moodle.org/>) for interacting with their peers and their tutors.

The paper introduces the CSCL concepts that constituted the theoretical basis for the experiment and then presents the actual learning scenarios. Then, we discuss the results together with a part of the tools that were developed for monitoring and analyzing the results produced by the community. Finally we present our conclusions together with a small analysis of the feedback received from the participants and our plans for further development.

2. The Learning Scenario

The experiments discussed in this paper were part of the Human-Computer Interaction course, at the Computer Science and Engineering Department of the “Politehnica” University of Bucharest, which is available for students that are in their final year of undergraduate studies. There are a lot of students that enroll for this course, for example 160 students in the 2007-2008 academic year and 324 students in the 2008-2009 one. During this course the students were taught the theory of interface design and evaluation, usability, user’s psychology and they had to apply their theoretical knowledge in practice by individually developing some web applications for the laboratory.

The standard learning scenario when we started the experiments was one in which students were learning the theoretical aspects at the course and were assigned non-differentiated homework exercises which asked them to use web standards and technologies discussed at the laboratory. The results were really good, as the students were glad to learn interesting theoretical details about interface design and also to use new state of the art technologies for developing web applications. However, the team of teachers and tutors felt that there was a place for improving the following aspects:

- Because the students were in their senior year, they were allowed to work with their favorite programming languages and technologies. Due to this characteristic, the number of technologies (APIs, tools, libraries) with which we actually worked during the lab was very large and the documentation was often sparse.
- The technologies were changing very fast (for example the APIs for interacting with social web sites like delicious.com) and they were not the same from one year to another. Moreover some of them were in a beta development phase and had different types of bugs that the students had to tackle.
- There were also a rather large number of types of applications that the students were able to choose from, in order to have a higher degree of personalization for the assignments.
- We felt that the students could benefit more if they worked in teams and, if possible, in ways that are close to a company environment, as they were in the last year of studies.

In order to improve these aspects, we proposed the following learning scenario, based on the Lave & Wenger’s [6] idea of community of practice. The students were divided in groups of four and were asked to construct a web site for their team using the technologies and the theoretical principles learnt in the course. Also they were asked to use a blog to share their personal experiences

with the technology and to discuss about the new and interesting details that they discovered during this course. Finally, the students were also asked to use chat for debates on a number of themes proposed by the teacher. One of the main ideas of our new learning scenario was that students can understand better how to design and implement Web 2.0 interaction techniques if they are also intensively using them for group knowledge building. An interesting starting point for this approach was the Virtual Company [7] learning scenario. In the Virtual Company learning scenario the students work in project teams and develop competences by collaboratively engaging in real tasks from real customers. The Virtual Company scenario is very good for our senior students, but we needed to adapt it because we were not able to use real customers and tasks for such a large number of students. Therefore we asked the students to consider their teams as their own virtual companies and their assignments as their company's projects.

As mentioned above, the new leaning scenario includes another way of social building of knowledge in small groups: instant messenger conferencing. Students had 1-3 such assignments and for each of them they were assigned a topic to study and, afterwards, to debate in chat sessions in groups of 3 to 8. For example, the topic of one of the assignment was a comparison between different social web technologies used for collaboration: wikis, blogs, discussion forums and chat conferences. After studying individually this topic using online and offline documents, the students formed teams to discuss the subject over chat: in the first part of the conversation, each student had to support one of these technologies by presenting its features and advantages and criticize the others by invoking their flaws. In the final part of the chat, they also had to discuss how they could integrate all these technologies into a single online collaboration platform. Thus, in a single conversation the students firstly had to engage into a debate those outcomes are then used for building collaboratively a solution to a given problem. The ConcertChat environment [8] was used for this task due to its collaborative features, such as the use of explicit references and of a whiteboard and because it saved the chat conversations persistently on the server. These chat logs are then used by the tutors to evaluate the results of the chat conversations and the performance of the students.

3. Monitoring and evaluation tools

Considering the large amount of students that we worked with, we had to find a mechanism to monitor and to evaluate semi-automatically their activities. We also needed a mechanism to take advantage of the knowledge artifacts that have been produced by the students through blogs. Therefore, we have designed an application that is used for indexing and classifying the content from the distributed knowledge base represented by the blogs of the students. The

automatic classification of the blog posts is also useful for the evaluation of the quality of a blog. Moreover, it was very important to develop tools that assist the tutors in the analysis of the chat conferences, as this task proved to be very complicated and time-consuming. The results of using these newly-developed tools are presented in the next section.

3.1. Chat evaluation tools

The tutors reported that the most difficult task was to evaluate the collaborative chat sessions, and the difficulty of this process increased proportionally to the number of participants and the length of the discussion (in each of the last 4 years, more than 150 chat sessions lasting at least one hour were performed). The evaluation should consider two distinct components: assessment of the degree of collaboration and involvement in the chat and determining the quality of the utterances issued by a participant in the conversational context. Both aspects are complex to measure without computer assistance, especially the latter as it needs to take into account several previous utterances that make up the context of the current one. Moreover, this evaluation made at an utterance level must be quantified and summed up for all the utterances of each participant, in order to assess the quality of the conversation. It is important to notice that the difficulty of the assessment of a collaborative chat also arises due to the fact that the conversation between more than 4 participants is different than a normal face-to-face conversation due to the fact that there are a number of different discussion threads that inter-animate at any point during the chat [5].

In order to improve the quality of the evaluation process for chats and to reduce the amount of time needed by the tutors, two different tools were developed to support them in this activity: Polyphony Analysis [5] and ASAP [9]. The former was first tested in 2007 and enhanced during the last two years, while the latter is a newer system that dates from mid-2008. Both employ natural language processing and social network analysis techniques, but they use different approaches. Polyphony Analysis uses Bakhtin's ideas of voices and polyphony [10] in order to discover implicit references between the utterances of the chat, thus constructing discussion threads. Discourse analysis techniques including cue phrases and speech acts detection were used in order to achieve this task. Moreover, temporal constraints and semantic similarities based on the WordNet lexical ontology were also utilized. The same task has been undertaken using different methods by Wang [11] for newsgroup conversations and Shen [12] for dynamic text message streams including instant messaging. The system has also other powerful features that guide the tutors, the most important of these are: extraction of the most important topics in the chat using a grouping algorithm based on sets of synonyms, an evaluation module that assigns a value to each utterance in the chat and based on these values it provides a score for each

participant and an interactive component that permits inspecting the utterances and references in the conversation. The evaluation module uses both NLP (Natural Language Processing) techniques like keyword and speech acts identification, semantic similarities and SNA (Social Networks Analysis) indicators like the number and type of input and output links in the discussion graph. ASAP (An Advanced System for Assessing Chat Participants) uses the social network of the participants, the utterances graph which result from the succession of turns in the chat and the explicit references between them. Thus, the social network can be modeled as a graph that has the participants as vertices and references between utterances as edges. In order to evaluate the competency of each participant several qualitative and quantitative factors are being considered such as the characteristics of the social network (e.g. closeness, rank), the speech acts that are present in each utterance and the importance of the words in each utterance. The next version of ASAP is called ChAMP (Chat Assessment and Modeling Program) and is used as an evaluation tool designed for automatically grading a participant for his contributions in a chat environment based on social networking factors and a semantic approach by using LSA (Latent Semantic Analysis) for assessing the importance and relevance of each utterance related to the specific domain of each conversation. Preliminary results for the automatic grading offered by the two systems and the overall time improvements for the tutors that used them are presented in the next section.

3.2. Blog analysis

Considering the huge number of blogs and posts that were produced, the rather small number of teaching assistants and the time constraints for each assistant, we needed to develop an assistance tool for the evaluation of the blogs. For this reason we have imposed that the blogs use a common platform – *Blogspot.com* in our case. We have chosen it because it uses Atom [13] for syndicating posts and comments so all we needed was to develop a tool that periodically gathered the feeds from every blog and indexed them in a database. This database contained all the data necessary to analyze the content of the students' articles and to apply clustering algorithms in order to group them on different topics of interest. In 2008, the database consisted of more than 500 posts which contained more than 200 words with a frequency greater than 50 (ignoring the stop words). The content was more than double in 2009. Using this database we have also tried to automatically identify concepts from the computer science domain, especially those related to the course: web standards and technologies, interface design and usability and more. As most of these terms don't appear in regular linguistic ontologies such as WordNet [14], we have used a public knowledge base called Freebase [15] for identifying computer science-related terms. The terms were searched using the Freebase web services and Freebase's

own query language MQL [15]. The query tried to identify if the term had one of the following standard Freebase domains/types: {computer, software, internet}. The script had a success rate of over 80%, thus managing to identify most of the common used abbreviations in the blog posts such as the popular web standards and technologies like “RDF”, “RSS”, “XML”, etc.

The next section focuses on the description of the clustering algorithms that were employed on this blog data in order to provide support to the tutors during the evaluation process. The most important results and conclusions on using this analysis tools are discussed in section 4.

3.3. Clustering algorithms

In our context, the clustering algorithms were used to group together the posts from all the different blogs that discuss similar subjects. As described in the next section, this was an important step for the evaluation of the content created on the blogs.

Clustering algorithms are used for the unsupervised partitioning of large datasets into clusters, which are groups of similar objects. The partitioning is based on a set of features that allow differentiating the objects, such that “to place similar objects in the same group and to assign dissimilar objects to different groups” [16]. The clustering algorithms can be classified in hierarchical and flat (non-hierarchical) clustering according to the structures they produce. Flat clustering algorithms consist of a number of clusters having no defined relation between them, while hierarchical clustering algorithms create a dendrogram of clusters, with the meaning that a child represents a subclass of its parent. Another classification criterion is whether algorithms allow objects to be included in a single (hard clustering) or in more clusters at the same time (soft clustering). Usually, hierarchical clustering uses the ‘hard’ cluster assignment method and therefore these algorithms are not suitable for many problems in natural language processing, where most of the objects could be part of more clusters at the same time. The flat clustering algorithms can use both the hard and the soft methods. The clusters are usually created in an iterative manner, starting with the generation of a number of random seeds that will be the starting points (the centroids) in creating the clusters – each seed will lead to a cluster. Then, each instance is assigned to the cluster that has its centroid the closest to this instance. After all the instances have been assigned to clusters and the first partition is obtained, the centroids are refined in order to represent the center of the newly created clusters more accurately. Then, a new iteration starts by re-assigning the instances to the clusters represented by the new centroids. This process repeats until the improvement in the clusters of data goes below a threshold or when an imposed number of iterations has been reached. The simplest flat clustering algorithm is the K-means algorithm and usually is the first one to be applied on a new dataset

since it is a very efficient algorithm and its results are often adequate. This algorithm assumes a simple Euclidean representation space, and uses either the Euclidean distance or the Norm distance to compute the distance between an instance and the centroid of the cluster. When the Euclidean space assumption fails, another algorithm can be used, for example, the EM algorithm [17].

We have used the K-means algorithm to create clusters of posts following the main topics of the course and laboratory classes. Having the posts grouped, it was easier for the teacher to grade them by comparing it to the content of the whole cluster, to the content of the other posts from the same cluster or to the teaching material used in class. It was also easier to discover if there were posts that have been copied from each other or from other sources on the internet by discovering duplicate or very similar content on different blogs.

4. Main Results of the Learning Experiment

We consider that the most important result from these experiments is the actual collaborative knowledge constructed by the groups of students. The resulted content and the connections that were created between students are very valuable for many reasons: documenting new technologies and less known problems, creating bonds between students and pinning down actual knowledge.

We have analyzed the types of content created in order to establish their actual value for the community and we have also tried to evaluate which was the actual educational gain for the individual student from this exercise.

We had 45 blogging teams in 2007-2008 and 96 ones in 2008-2009. These teams produced over 700 posts in the first year and 2200 in the second one. This content was not always of top quality and neither was it always useful, so we needed to define some criteria for interesting and useful posts and blogs.

The first thing we did was to differentiate the blogs according to the quantity of the content. The number of posts per blog, evolved from a minimum of 4 posts to a maximum of 130 for a 3 months time span, with an average of 16.5 posts in the first year and of 23.6 posts in 2008-2009. The median was very close to the average in both cases (16, respectively 23 posts per blog). Then, we looked at the type of each post. After clustering the posts, we have classified each cluster into 4 categories – personal experiences (bug solving, experience with a new API or with creating a new feature for a given API), studied technologies (descriptions or personal views on technologies studied at the course or lab), new technologies (also descriptions and/or views of new technologies that are usually based on technical documentation more than on personal experience), and miscellaneous (project's journal, news, social, fun, and other articles meant to attract visitors and develop social interactions between the teams). Probably, due to a miscommunication between the tutors and the students or just because the

students were not used to working at a task that involved blogging, the “miscellaneous” category contained around 48% of the posts in the first year. It is important to notice that the number of posts in this category severely decreased to just 28% in the second year, due to the lessons the tutors learnt by using the clustering and classification tools. The main reasons for the high number of posts in this category are: they are written more easily as they do not require a lot of expertise and the students have observed that their colleagues react more with comments and future visits to this kinds of posts and they chose the easy way to increase the number of visitors. In the 2008-2009 edition, we emphasized the fact that the content posts are the most important, thus the ratio of the misc posts dropped to almost half of the initial one. Due to the large number of posts in this category we have decided to eliminate it from the analysis of the distribution of the posts in the content categories that in shown in the table below.

Table 1

Distribution of posts according to their categories

Percentage of posts in each category	Personal Experiences	New Technologies	Studied Technologies
2008 (%)	23	43	34
2009 (%)	18	21	61

We considered that the most useful posts were the ones that presented personal experiences as they are much more valuable from the point of view of the community and of the individual user. The percentage of these posts was quite smaller than our expectances, especially in the second year. Only 18% of the valuable content posts represented personal experiences, but these however represent around 300 very relevant short articles that make up an important knowledge base for the next editions of the course. From the evaluation point of view, the teams that had an important number of personal experiences posts were graded better than the ones that had less.

Another important category was the one of new technologies. These articles presented ideas or technologies that were connected to the content of the course or of the lab but there was not enough time to study them. Such examples are Microsoft’s Silverlight (<http://silverlight.net>) for implementing interfaces, FOAF (<http://www.foaf-project.org/>) for semantically describing networks of people, SMIL (<http://www.w3.org/TR/REC-smil/>) as an XML format for describing presentations and so on. The articles presenting technologies not connected to the course were considered part of the misc category. This category is also interesting from the perspective of the results. Although 43% of the posts from the first year were in this category, their number decreased to just 21% in the second one. This happened due to the fact that some concepts and technologies were reconsidered by the teaching team after the feedback and articles of the first year and were integrated in the lab for the second year. This shows the iterative

nature of this process and the fact that the teaching team also receives valuable help using this scenario, especially because the students are in the senior year and have a high volume of practical knowledge. The studied technologies category contained 34% of the content posts in 2008 and 61% of the content in 2009. While these numbers confirm the affirmation that some of the new technologies in 2008 were present in the 2009 lab material, we consider that these posts are the least valuable of all. From the point of view of the actual content, most of these posts did nothing more than restating facts from the official learning materials and bibliography, some using personal examples and some using just the official examples.

One of the most eloquent examples of the benefits offered by using this scenario is the following. CTT [18], an application that we have used in the laboratory for designing a user interface, had a bug which made it difficult for the students to perform the given assignment. As the application was a research one and not very well documented, the bug was hard to overcome and it wasted a lot of time to quite a lot of students. Finally, some of them solved it and wrote the solution on their team's blog, documenting the error and how it can be solved. The solution spread very rapidly between students in all the teams and many spent less time with the application bug and more time on actually getting the homework done. Also, thanks to that post on the blog, the solution to this problem will be presented to the students in the year to come. This example shows the benefit of using personal experiences as blogging materials for school projects, the advantages of organizing a knowledge building community of students and explains why blogs can be used with success for communication inside such a community.

The first time when the two systems designed for chat analysis were used by the tutors as support for the evaluation process was during the 2008-2009 academic year [19]. The tutors used the statistics that the tools provided in order to have a better understanding of a chat conference in less time. For example, the implicit references, the threads and the list of important topics were used for enhancing the content-based analysis. The same is available for the scores provided by Polyphony Analysis for each utterance in the chat based on their content and on speech acts factors such as approval and disapproval. Furthermore, the total score of all the utterances issued by a participant is used to provide an automatic grading. ASAP also provides several statistics and a grading by taking into account the most frequent concepts and the social network. By using these statistics as well as the visualization from Polyphony the tutor can better evaluate the degree of involvement of each student in the chat. The preliminary tests were conducted on a group of 4 chat sessions involving teams of 4 members each that were analyzed separately by four tutors – two of them using the tools and the other two without any software assistance. The grading error for Polyphony is

10.1%, twice better than ASAP, and quite close to the error rates of the tutors [19]. Nevertheless, the correlation between the average tutor grade and the grades provided by the two systems are significantly poorer compared to those of the tutors. Still, the correlation obtained by Polyphony Analysis is encouraging for a subset of three chats out of the four - .85, only slightly worse than the average tutor correlation. The improvement in time needed for the evaluation of a chat session is also hopeful as the time required for analysis was reduced by more than 30% for the tutors employing the analysis tools.

6. Conclusions

The work presented in this paper describes a complex learning experiment, involving collaborative technologies like blogs and chats. The experiment has been improved over several years and more than 700 students were part of it. These students had the opportunity to learn interesting things by integrating in a learning community and by creating knowledge artifacts for this community. The knowledge artifacts were used by the students in the same year and also, as the results have showed it, they had an influence over the students that followed the course in the next years. In order to validate this complex learning process we have developed analysis tools for chats and blogs. These tools helped the teachers by providing a preliminary assessment of the information presented by the students.

The results show that this way of learning was appreciated by the students who worked hard and produced an important quantity of valuable content. The students also provided valuable and positive feedback through the internal university feedback forms which are available online on Moodle and are anonymous. 33% of the students who took the course considered the project and applications part of the course to be very good (highest grade) and 45% considered it to be good (second highest).

Moreover, students' involvement in chat assignments was beyond our expectations. We suggested that a typical chat session would have about one hour length and often this duration was longer. Moreover, the students inter-animating in a high degree, examples of different such patterns being discussed in [4, 5].

Finally we consider that this paper proves that social web technologies can be used to supplement classical learning with good learning results and they offer a pleasant learning experience for the students. We also believe that this experiment can be successfully implemented in a large number of other courses in the context of higher education or lifelong learning.

REFERENCES

- [1] *Hoschka, P.*, CSCW research at GMD-FIT: from basic groupware to the social Web. SIGGROUP Bull. 19, 2 (Aug. 1998), 5-9.
- [2] *Vygotsky, L.*, Mind in Society, Cambridge, MA: Harvard University Press
- [3] *Stahl, G.*, Group cognition: Computer support for building collaborative knowledge. Cambridge, MA: MIT Press. 2006
- [4] *Trausan-Matu, S., Stahl, G., Sarmiento, J.*, Supporting Polyphonic Collaborative Learning, E-service Journal, vol. 6, nr. 1, Indiana University Press, 2007, pp. 58-74.
- [5] *Trausan-Matu, S., Rebedea, T.*, Polyphonic Inter-Animation of Voices in VMT, in Stahl, G. (ed.), Studying Virtual Math Teams, Springer, New York, 2009.
- [6] *Lave, J., Wenger, E.*, Situated Learning: Legitimate Peripheral Participation, Cambridge University Press. 1991
- [7] *Spoelstra, H., Posea, V., van Bruggen, J., & Koper, R.*, Modeling the Virtual Company Educational Scenario Competence Assessment in the Cooper environment. 1st International Workshop on Collaborative Open Environments for Project-Centered Learning, Crete, Greece, September 2007.
- [8] *Mühlplfordt, M., Wessner, M.*, Explicit Referencing in Chat Supports Collaborative Learning, in T. Koschmann, D. D. Suthers & T.-W. Chan (Eds.), Computer Supported Collaborative Learning 2005. The Next 10 Years! (CSCL 2005). Mahwah, NJ: Lawrence Erlbaum Associates (2005).
- [9] *M. Dascalu, E.V. Chioasca, S. Trausan-Matu*, ASAP - An Advanced System for Assessing Chat Participants, in D. Dochev, M. Pistore, and P. Traverso (Eds.): AIMS 2008, LNAI 5253 (2008), Springer, 58–68.
- [10] *Bakhtin, M.M.*, Problems of Dostoevsky's Poetics, Ardis (1973)
- [11] *Wang, Y. C., Joshi, M., Cohen, W. W., Rosé, C. P.*, Recovering Implicit Thread Structure in Newsgroup Style Conversations. In Proceedings of the 2nd International Conference on Weblogs and Social Media (ICWSM II), Seattle, USA (2008).
- [12] *Shen, D., Yang, Q., Sun, J.-T., Chen, Z.*, Thread detection in dynamic text message streams. In Proceedings of the Twenty-Ninth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 35-42). ACM: New York (2006)
- [13] The Atom Syndication Format - <http://www.ietf.org/rfc/rfc4287.txt> (Retrieved from the Internet March 2009)
- [14] *Miller, G.A.*, WordNet: a lexical database for English.' In: Communications of the ACM 38 (11), November 1995, pp. 39 - 41.
- [15] *Bollacker, K., Cook, R., Tufts, P.*, Freebase: A Shared Database of Structured General Human Knowledge. Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22-26, 2007, Vancouver, British Columbia, Canada. AAAI Press 2007, ISBN 978-1-57735-323-2 p. 1962-1963
- [16] *Manning, C., Schütze, H.*, Foundations of Statistical Natural Language Processing, MIT Press. Cambridge, MA: May 1999
- [17] *Dempster, A., Laird, N., Rubin, D.*, Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society, Series B, 39(1):1–38, 1977
- [18] *Mori, G., Paternò, F., Santoro, C.*, CTTE: Support for Developing and Analyzing Task Models for Interactive System Design, IEEE Transactions on Software Engineering (August 2002, pp.797-813).
- [19] *Rebedea, T., Trausan-Matu, S.*, Computer-Assisted Evaluation of CSCL Chat Conversations, Computer Supported Collaborative Learning 2009 (CSCL 2009)