Predicting Correctness of Problem Solving in ITS with a Temporal Collaborative Filtering Approach

Suleyman Cetintas, Luo Si — Department of Computer Sciences
Yan Ping Xin, Casey Hord — Department of Educational Studies
Purdue University, West Lafayette, IN, 47907, USA
{scetinta,lsi}@cs.purdue.edu {yxin,thord}@ purdue.edu

Abstract. Collaborative filtering (CF) is a technique that utilizes how users are associated with items in a target application and predicts the utility of items for a particular user. Temporal collaborative filtering (temporal CF) is a time-sensitive CF approach that considers the change in user-item interactions over time. Despite its capability to deal with dynamic educational applications with rapidly changing user-item interactions, there is no prior research of temporal CF on educational tasks. This paper proposes a temporal CF approach to automatically predict the correctness of students' problem solving in an intelligent math tutoring system. Unlike traditional user-item interactions, a student may work on the same problem multiple times, and there are usually multiple interactions for a student-problem pair. The proposed temporal CF approach effectively utilizes information coming from multiple interactions and is compared to i) a traditional CF approach, ii) a temporal CF approach that uses a sliding-time-window but ignores old data and multiple interactions and iii) a combined temporal CF approach that uses a sliding-time-window together with multiple interactions. An extensive set of experiment results show that using multiple-interactions significantly improves the prediction accuracy while using sliding-time-windows doesn't make a significant difference.

1 Introduction

Collaborative information filtering, or collaborative filtering (CF), is an important technology that utilizes how users of a system are associated with items in an application to predict the utility of items for a particular user. Specific type of items and associations differ by target applications (e.g., buying books at Amazon, reading news at Google, and renting CDs at Netflix, etc.). CF techniques have been applied mainly in many e-commerce systems for business purposes. Recently, they have also been used for educational applications such as legal argumentation [11], educational resource recommendation [14], writing skills training [5] and eLearning [9]. An important issue with many applications is that users' behaviors change over time and a static CF approach may not always be optimal to anticipate users' future behaviors. Temporal collaborative filtering (temporal CF) is a CF approach that adapts itself to the constantly changing system dynamics (i.e., user-item associations). Temporal CF has gained substantial interests in business applications such as movie and music track recommendation [1], [6], [7], [8]. However, despite its capability to deal with dynamic educational applications with rapidly changing user-item interactions, temporal CF has not been applied to educational tasks yet.

Table 1. Statistics about the number of times each worksheet is repeated.

Repetition Statistics	Worksheet Name										
	Equal Group				Multiplicative Compare				Mixed		
	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3
Mean	2.2	4	1.6	1.5	1.3	3.3	1.4	1.2	1.3	1.1	1
Std. Dev.	1.8	2	0.5	0.7	0.5	2.9	0.9	0.6	0.9	0.3	0

To the best of our knowledge there is no prior research of temporal CF on the automatic detection of whether a student will be able to correctly answer a question with a high-level student model (i.e., without using any expert knowledge of the domain). Prior research utilized combinations of features such as time, mouse tracking and performance related features [2], [4], [12]; most of which are extracted while a student is solving the problem. However, it is not possible to predict whether a student will be able to solve a problem before the problem is presented to the student (i.e., enough data is collected) which limits the utilization of the student model. For example, related prior research was not able to give an early feedback to the student depending on his likelihood to solve the problem or change the problem with an easier or harder one. Prior work on temporal CF focused on business applications such as movie [6], [7], [8] and music track recommendation [1]. A simple and popular approach in the prior work of temporal CF (to deal with rapidly changing user-item associations) is the utilization of sliding time windows, which uses new data in a current sliding window and discards (or assigns decreasing weights on) old data [6][13]. However, unlike a traditional CF based application (e.g., a user votes for a movie only once), a student has multiple interactions with a task/problem in a problem solving environment and this data is ignored by traditional CF approaches or temporal CF approaches such as sliding window.

This paper proposes a novel temporal CF approach that can automatically predict the correctness of students' problem solving in an intelligent math tutor by utilizing the information coming from multiple interactions. The new approach is compared to i) a traditional CF approach, ii) a temporal CF approach that uses sliding time window but ignores old data and multiple interactions and iii) a novel hybrid temporal CF approach that uses a sliding-time window together with multiple interactions. We show that using multiple interactions significantly improves the prediction accuracy. It is also shown that although using sliding time window has been shown to be effective in prior research [6], [13]; it doesn't make a significant difference in this work. Furthermore, a novel hybrid approach combining the proposed temporal CF together with the temporal CF using sliding time windows is found to be not significantly different than using the proposed temporal CF approach of only utilizing multiple interactions. Finally, the Pearson Correlation Coefficient (PCC) method is found to be not significantly different than the Vector Similarity (VS) while calculating the similarity between two users, although PCC has been shown to be more effective than VS in other applications (e.g., in business domain) [3].

2 Data

Data from a study conducted in fall 2008 and partly in spring 2009 in a nearby elementary school was used in this work. The study was conducted in mathematics classrooms using a math tutoring software (that has been developed by the authors). The tutoring software teaches problem solving skills for Equal Group (EG) and Multiplicative Compare (MC) problems. These two problem types are a subset of the most important mathematical word problem types that represent about 78% of the problems in a fourth grade mathematics textbook [10]. In the tutoring system; first, a conceptual instruction session is studied by a student followed by problem solving sections to test their understanding. Both of conceptual instruction and problem solving parts require students to work one-on-one with the tutoring software and if students fail to pass a problem solving session, they have to repeat the corresponding conceptual instruction and the problem solving session. Details about the number of repetitions of each worksheet (by all students) are given in Table 1. Space limitations preclude discussing in detail but the tutoring software has a total of 4 conceptual instruction sessions and 11 problem solving worksheets that have 12 questions each (4 for Equal Group worksheets, 4 for Multiplicative Compare worksheets, 3 Mixed worksheets each of which include 6 EG & 6 MC problems). The software is supported with animations, audio (with more than 500 audio files), instructional hints, exercises etc.

The study with the tutoring system included 10 students among which 3 students have learning disabilities, 1 student has emotional disorder and 1 student has emotional disorder combined with a mild intellectual disability. Students used the tutor for several class sessions of 30 minutes (on average 18.7 sessions per student with standard deviation of 3.23 sessions) during which their interaction with the tutoring system was logged in a centralized database. A total of 2388 problems (corresponding to a total of 199 worksheets) were solved with 1670 problems correctly solved (with average 167.0 and std. 23.1) and 718 incorrectly solved (with average 71.8 and std. 32.5). Data from 9 students was used as training data to build the models for making predictions for the remaining 1 student (who is used as the test data) at each configuration. That is, all 10 students are used as the test data alternatively and the data from other 9 students is used as training data. The averages of the results for all configurations are reported.

3 Methods and Modeling Approaches

3.1 Collaborative Filtering Framework

Predicting correctness of problem solving with a high-level student model (i.e., without using any expert knowledge of the domain) can be treated as a collaborative filtering problem, which models students' likelihood to solve problems. The collaborative filtering framework in this work can be defined as follows: assume there are M students and L worksheets in the system, where each of the worksheets has K questions. Note that M is 10, L is 11 and K is 12 in this work. Let w_m^l be the l^{th} worksheet by m^{th} student, $r(w_m^l)$ be the total number of repetitions of l^{th} worksheet by m^{th} student, $w_m^{l,n}$ be the n^{th} repetition of l^{th} worksheet by m^{th} student, $w_m^{l,n}(i)$ be the i^{th} problem in that worksheet and $s(\,w_m^{l,n}(i)\,)$ be the student's score on the problem (i.e., 1 means the student solved the problem correctly and 0 means the student solved the problem incorrectly). A first step in CF is to calculate the similarities between students. There are two common techniques for this task: Pearson Coefficient Correlation (PCC) and Vector Similarity (VS). PCC can be calculated as follows:

$$Sim(u,u^t) = \frac{\sum_{l=1}^{L}\sum_{k=1}^{K}(s(w_u^l(k)) - \overline{s(w_u)})(s\left(w_{u^t}^l(k)\right) - \overline{s(w_{u^t})})}{\sqrt{\sum_{l=1}^{L}\sum_{k=1}^{K}(s(w_u^l(k)) - \overline{s(w_u)})^2}\ \sqrt{\sum_{l=1}^{L}\sum_{k=1}^{K}(s\left(w_{u^t}^l(k)\right) - \overline{s(w_{u^t})})^2}} \tag{1}$$

where $Sim(u,u^t)$ is the similarity score between students u and u^t, $\overline{s(w_u)}$ is the average score of student u on all problems. The vector similarity (VS) can be calculated as follows:

$$Sim(u,u^t) = \frac{\sum_{l=1}^{L}\sum_{k=1}^{K} s(w_u^l(k))\, s\left(w_{u^t}^l(k)\right)}{\sqrt{\sum_{l=1}^{L}\sum_{k=1}^{K} s(w_u^l(k))^2}\ \sqrt{\sum_{l=1}^{L}\sum_{k=1}^{K} s\left(w_{u^t}^l(k)\right)^2}} \tag{2}$$

After the similarity between users are calculated, prediction for a problem can be computed by using the sum of the scores of training users on the problem weighted by the similarity between users as follows:

$$s\left(\widehat{w_{u^t}^l(k)}\right) = \overline{s(w_{u^t})} + \frac{\sum_{u=1}^{M} Sim(u,u^t)(s(w_u^l(k)) - \overline{s(w_u)})}{\sum_{u=1}^{M}|Sim(u,u^t)|} \tag{3}$$

Note that the above CF approach ignores the multiple interactions (i.e., repetitions of worksheets) and only uses the latest scores of a student on a worksheet (e.g., in the same way it uses the latest rating of a user on a movie). Therefore the repetition index, that should be included as follows $s\left(w_u^{l,r(w_u^l)}(k)\right)$ for a problem $s(w_u^l(k))$, is omitted in the formulas for simplicity.

To predict the correctness of problem solving for a test worksheet of a student, all of the previous worksheets (i.e., the worksheets that are already solved and available for use) of that student are used in this modeling approach.

This modeling approach will serve as a baseline and will be referred as Mod_Baseline_All.

3.2 Temporal Collaborative Filtering with Sliding Time Window

Temporal collaborative filtering is a time-sensitive CF approach that adapts itself to the rapidly changing user-item interactions. The user-item associations in many applications change with time as user's behavior can change over time. One of the simple temporal CF approaches to deal with changing dynamics is to favor newer data than older data for having an up-to-date model of the users' behavior. Sliding time windows (and their variants) that only use the newest data is a popular approach, which has been followed in prior research [6], [13]. To the best of our knowledge, this is the first work utilizing sliding time windows on an application in the education domain. In this work, the sliding time window is defined as using the last x worksheets for $x \in \{1,2,3,4\}$. Sliding time window approach will be referred as Mod_Baseline_Window.

3.3 Temporal Collaborative Filtering with Multiple Interactions and a Hybrid Approach

Unlike traditional user-item interactions, students can work on a problem several times. In most CF based systems, users don't interact with the items multiple times, therefore a CF approach or a temporal CF approach don't take into account the information coming from multiple interactions that happen in educational environments such as tutoring systems with problem solving activities. To predict a student's performance on a problem, the use of the student's past performance on the same problem is a valuable source of data that should not be ignored. By utilizing this data, it becomes possible not only to compare the latest performances of students' on other problems (like in CF or temporal CF approaches) but also to compare the learning curves by comparing their first, second, etc. trials on the same problems. In this work, the temporal CF approach that utilizes multiple interactions can be calculated with the following changes over the VS and prediction formulas as follows: the new vector similarity (VS):

$$Sim(u,u^t) = \frac{\sum_{l=1}^{L}\sum_{n=1}^{r(w_{u^t}^{l})}\sum_{k=1}^{K} s\left(w_u^{l,n}(k)\right) s\left(w_{u^t}^{l,n}(k)\right)}{\sqrt{\sum_{l=1}^{L}\sum_{n=1}^{r(w_{u^t}^{l})}\sum_{k=1}^{K} s\left(w_u^{l,n}(k)\right)^2}\sqrt{\sum_{l=1}^{L}\sum_{n=1}^{r(w_{u^t}^{l})}\sum_{k=1}^{K} s\left(w_{u^t}^{l,n}(k)\right)^2}} \tag{4}$$

So, if the test student is working on worksheet $w_{u^t}^{l}$ for the $\left(r\left(w_{u^t}^{l}\right)+1\right)^{\text{th}}$ time, her/his 1st trial on the worksheet is compared with other students' first trial on the same worksheet, her/his second trial is compared with others' second trial, and her/his $r\left(w_{u^t}^{l}\right)^{\text{th}}$ trial is compared with other students' $r\left(w_{u^t}^{l}\right)^{\text{th}}$ trial. In the traditional approach only the latest trials of the test student on previous worksheets (i.e., $l_{prev} < l$ and $w_{u^t}^{l_{prev},r\left(w_{u^t}^{l_{prev}}\right)}$) is compared with only the latest trials of other students on those worksheets (i.e., $w_{u}^{l_{prev},r\left(w_{u}^{l_{prev}}\right)}$). An important thing to note is the dimension mismatch problem that can happen when two students have different number of trials on a worksheet which may cause $r\left(w_{u^t}^{l}\right) > r(w_u^l)$ where there is no corresponding trial of worksheet w_u^l from student u to compare with student u^t. In such a case, the approach used in this paper is as follows: $w_u^{l,n}(k) = w_u^{l,r(w_u^l)}(k) \quad for\ n > r(w_u^l)$. That is, for instance, if the third trial of a test user's worksheet is being compared and a training user has repeated that worksheet only twice, her/his third (or more) trial is assumed to be the same with her/his second (i.e., last) trial. This approach is better than just comparing $r(w_u^l)^{\text{th}}$ trials only, for two reasons: i) as the difference $r\left(w_{u^t}^{l}\right) - r(w_u^l)$ becomes bigger, the proposed approach punishes the similarity more (and since similar students should have similar repetition behaviors this should be the case) and ii) students solve a worksheet until they master the worksheet and get enough score on the worksheet and therefore their last trial is a good representation of their final status (after their last trial) on that worksheet.

After the new similarity between users is calculated, the new prediction formula becomes:

$$s\left(\widehat{w_{u^t}^{l,r\left(w^l_{u^t}\right)}(k)}\right) = \overline{s(w_{u^t})} + \frac{\sum_{u=1}^{M} Sim(u,u^t)\left(s\left(w_u^{l,r\left(w^l_{u^t}\right)}(k)\right) - \overline{s(w_u)}\right)}{\sum_{u=1}^{M} |Sim(u,u^t)|} \quad (5)$$

So to predict the $r\left(w^l_{u^t}\right)^{\text{th}}$ repetition of a worksheet of a test user, only the data from $r\left(w^l_{u^t}\right)^{\text{th}}$ repetitions of training students are used, rather than using the data from their final (i.e., $r(w^l_u)^{\text{th}}$) repetitions. This modeling approach will be referred as Mod_MultInt_All.

To better evaluate the effect of sliding time windows approach, we also propose a novel hybrid approach combining the proposed temporal CF (i.e., Mod_MultInt_All) together with the temporal CF approach that uses sliding time windows. This hybrid temporal CF approach will be referred as Mod_MultInt_Window.

4 Experimental Methodology: Evaluation Metric

Mean Absolute Error (MAE) has been used as a popular evaluation metric in the prior work [1], [3], [6]; and is calculated by looking at the mean absolute deviation of a test student's predicted scores from her/his actual score on the problems. In this work, the MAE of a test worksheet $w_u^{l,n}$ (i.e., the n^{th} repetition of l^{th} worksheet by m^{th} student) can be computed as:

$$MAE\left(w_u^{l,n}\right) = \frac{\sum_{k=1}^{K} \left| s\left(w_u^{l,n}(k)\right) - s\left(\widehat{w_u^{l,n}(k)}\right) \right|}{K} \quad (6)$$

where K is the number of problems in a worksheet, $s\left(\widehat{w_u^{l,n}(k)}\right)$ is student's predicted score on the problem (i.e., 1 if predicted to be correctly solved, 0 otherwise) and $s\left(w_u^{l,n}(k)\right)$ is student's actual score (i.e., 1 or 0) on the problem.

Note that while predicting a test worksheet for a user, all of the previous worksheets or part of them (i.e., in the sliding window approach) are used for calculating the similarity between users. For instance, if the 3rd repetition of MC worksheet 2 is the test worksheet; then Mod_Baseline_All will use only the final repetitions of all previous worksheets (i.e., last repetition of EG worksheets 1,2,3,4 and MC worksheet 1). On the other hand Mod_MultInt_All will use the previous repetitions of the test worksheet together with all repetitions of all previous worksheets (i.e., all repetitions of EG worksheets 1,2,3,4 and MC worksheet 1 together with 1st and 2nd repetitions of MC worksheet 2). Sliding window versions of both approaches only utilize the last k of the training worksheets (explained above) depending on the window size. Therefore each worksheet is predicted separately and the MAE is calculated for each predicted worksheet of a student separately. The average of the MAEs for all test worksheets of a student (i.e., all worksheets except

Table 2. Results of the Mod_Baseline_All, Mod_Baseline_Window, Mod_MultInt_All and Mod_MultInt_Window CF approaches in comparison to each other for two similarity configurations: i) Pearson Correlation Coefficient (PCC) and ii) Vector Space (VS) and for four window sizes. The window size is the number of past worksheets used for calculating the similarity between students. The performance is evaluated with the MAE.

Methods			Similarity Metric: Pearson Correlation Coefficient (PCC)	Similarity Metric: Vector Space (VS)
Mod_Baseline_All			0.309	0.308
Mod_Baseline_Window	Window Size	1	0.296	0.306
		2	0.319	0.309
		3	0.296	0.306
		4	0.296	0.306
Mod_MultInt_All			0.268	0.269
Mod_MultInt_Window	Window Size	1	0.265	0.286
		2	0.275	0.285
		3	0.270	0.282
		4	0.275	0.283

the first worksheet, namely EG worksheet 1) is the MAE of that student. The mean of the MAEs of all students is the final MAE; and this final MAE is reported in this work.

5 Experiment Results

This section presents the experimental results of the methods that are proposed in Methods and Models section. All the methods were evaluated on the dataset as described in Data section (i.e., Section 2).

5.1 The Performance of Temporal CF with Sliding Time Window (i.e., Mod_Baseline_Window)

The first set of experiments was conducted to evaluate the effectiveness of the temporal CF approach of using sliding time windows. More specifically, Mod_Baseline_Window and Mod_MultInt_Window CF approaches are compared to Mod_Baseline_All and Mod_MultInt_All CF approaches (details about which are given in Section 3) with each other on the prediction of problem solving task for four different window sizes. Their performance can be seen in Tables 2. It can be seen that the Mod_Baseline_Window approach slightly outperforms Mod_Baseline_All approach for window sizes 1, 3 & 4; and Mod_Baseline_All approach slightly outperforms Mod_Baseline_Window approach for window size 2. Similarly the Mod_MultInt_Window approach slightly outperforms Mod_MultInt_All approach for window size 1 (with PCC) and Mod_MultInt_All approach slightly outperforms Mod_MultInt_Window approach for window sizes more than 2 (and window size 1 with VS). Paired t-tests have been applied for this set of experiments and the improvement gained by using the sliding-time window or using different window

Table 3. Results of the Mod_MultInt_All and Mod_MultInt_Window CF approaches are shown in comparison to Mod_Baseline_All and Mod_Baseline_Window CF approaches for two similarity configurations: i) Pearson Corrleation Coefficient (PCC) and ii) Vector Space (VS). The performance is evaluated with the MAE.

Methods	Similarity Metric	
	Pearson Correlation Coefficient (PCC)	Vector Space (VS)
Mod_Baseline_All	0.309	0.308
Mod_MultInt_All	0.268	0.269
Mod_Baseline_Window	0.296	0.306
Mod_MultInt_Window	0.266	0.286

sizes has been found to be not significant (i.e., p-value is more than 0.01) in most of the configurations (there are some significant differences in favor of and against using time windows at the same time, so it is not possible to see a consistent and significant dominance of either approach over each other). To the best of our knowledge this is the first work using sliding time windows for an educational application (specifically for predicting the correctness of problem solving). Results discussed above show that, sliding time windows (or their variants), despite their positive effect on the applications of temporal CF over business applications [6], [13]; should be carefully considered.

5.2 The Performance of Temporal CF with Multiple Interactions and the Hybrid Approach (i.e., Mod_MultInt_All and Mod_MultInt_Window)

The second set of experiments was conducted to evaluate the effectiveness of the temporal CF approach of using multiple interactions and the hybrid temporal CF approach of using multiple interactions together with the sliding time windows. More specifically, Mod_MultInt_All and Mod_MultInt_Window (with window size 1) CF approaches are compared to Mod_Baseline_All and Mod_Baseline_Window (with window size 1) CF approaches (details about which are given in detail in Section 3) with each other. Their performance can be seen in Table 3. It can be seen that both of Mod_MultInt_All and Mod_MultInt_Window approaches significantly (with p-value less than 0.001) outperform Mod_Baseline_All and Mod_Baseline_Window approaches respectively. This explicitly shows that utilizing the information coming from the multiple interactions (i.e., repetitions of worksheets) is a much better approach than the default CF approach of using the latest user-item interactions for predicting correctness of problem solving.

The hybrid approach of utilizing sliding time window together with multiple interactions (i.e., Mod_MultInt_Window) has not been found to be significantly different than Mod_MultInt_All approach (i.e., p-value is more than 0.01). This is consistent with the results reported in the previous section. To better see the robustness of the Mod_MultInt_All approach, the average of PCC and VS results of the Mod_MultInt_All approach is shown in comparison to Mod_Baseline_All approach in Table 4 for all the test worksheets (i.e., all worksheets except EG worksheet 1). It can be seen that Mod_MultInt_All approach is robust across all worksheets and performs consistently better than the Mod_Baseline_All approach almost for all worksheets. It should also be noted that Mod_MultInt_All achieves

Table 4. Results of the Mod_Baseline_All and Mod_MultInt_All CF approaches in comparison to each other for all test worksheets (i.e., for all worksheets except the 1st worksheet, namely EG Worksheet 1). The performance is evaluated with the MAE and the average of PCC and VS based results are reported.

Methods	Test Worksheets									
	Equal Group			Multiplicative Compare				Mixed		
	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3
Mod_Baseline_All	0.46	0.35	0.38	0.13	0.31	0.28	0.39	0.34	0.18	0.28
Mod_MultInt_All	0.44	0.30	0.34	0.09	0.29	0.24	0.29	0.27	0.21	0.21

comparable performance with prior work on the prediction of problem solving with F_1 scores of 0.780 (for PCC) and 0.778 (for VS) [2], [4].

5.3 The Effect of Using Different Similarity Metrics (i.e., PCC and VS)

The third set of experiments was conducted to evaluate the effectiveness of the two popular similarity metrics: i) Pearson Correlation Coefficient (PCC) and ii) Vector Similarity (VS). It can be seen in Tables 2 and 3 that although the PCC seems to perform slightly better than the VS, the difference in their performance is not found to be significant (i.e., p-value is more than 0.01) for most cases. This is different from the prior work of CF in business domain, where PCC has been shown to perform better than VS [3]. This difference can be explained by the fact that in prior work such as movie recommendation, users' have different voting behaviors (i.e., some users tend to vote higher for all movies and some tend to vote lower). PCC can better deal with this user bias. Yet, in applications where user-item interactions are not user-voluntary such as this work; VS can perform comparable to PCC. [3]

6 Conclusion and Future Work

This paper proposes a novel temporal CF approach to predict the correctness of students' problem solving in an intelligent math tutor by utilizing the multiple interactions. Several modeling approaches with different configurations are studied for this application through extensive experiments. Empirical results show that a temporal CF approach utilizing the information coming from multiple interactions between student-problem pairs is much more effective than a CF approach that does not utilize this information. A temporal CF approach that uses a sliding time window is found to be not effective, although it has been shown to be an effective CF approach in business applications such as movie recommendation. Furthermore, a novel hybrid approach combining the proposed temporal CF together with the temporal CF using sliding time windows is found to be not significantly different than using the proposed temporal CF approach of only utilizing multiple interactions. Finally the common Pearson Correlation Coefficient similarity metric is found to be not significantly more effective than the common Vector Similarity, although PCC has been shown to be significantly more effective than VS in business applications such as movie recommendation [3].

There are several possibilities to extend the research. First, all of the CF approaches in this work are only using students' scores on the problems: it is possible to utilize problem features such as difficulty and readability levels. It is possible to design sophisticated CF algorithms that model the combination of temporal CF algorithms proposed in this work and problem features intelligently (e.g., via mixture models). Second, i) only 10 students and ii) data from a single application are used for evaluation in this work, more students and datasets from different applications can be used to assess the robustness of the proposed algorithms. Future work will be conducted mainly in those directions.

Acknowledgements: This research was partially supported by the NSF grants IIS-0749462, IIS-0746830 and DRL-0822296. Any opinions, findings, conclusions, or recommendations expressed in this paper are the authors', and do not necessarily reflect those of the sponsor.

References

1. Baltrunaz, L., Amatriain, X.: Towards Time-Dependant Recommendation based on Implicit Feedback. In Proceedings of the 3rd Int.al RecSys Conference Workshop 3: CARS (2009)
2. Beck, J.E., Woolf, B.P.: High Level Student Modeling with Machine Learning. In Proceedings of the 5th ITS Conference, p. 584-593 (2000)
3. Breese, J., Heckerman, D., Kadie, C.: Empirical Analysis of Predictive Algorithms for Collaborative Filtering. In Proceedings of the 14th UAI Conference (1998)
4. Cetintas, S., Si, L., Xin, Y. P. & Hord, C..: Predicting correctness of problem solving from low-level log data in intelligent tutoring systems. In Proceedings of the 2nd International EDM Conference, 230-239 (2009)
5. Cho, K., Schunn, C. C., Wilson, R. W.: Validity and Reliability of Scaffolded Peer Assessment of Writing From Instructor and Student Perspectives. In Journal of Educatonal Psychology, 98(4):891-901 (2006)
6. Ding, Y., Li, X.: Time Weight Collaborative Filtering. In Proceedings of the 14th ACM CIKM Conference (2005)
7. Koren, Y.: Collaborative Filtering with Temporal Dynamics. In Proceedings of the 15th International KDD Conference (2009)
8. Lathia, N., Hailes, S., Capra, L.: Temporal Collaborative Filtering with Adaptive Neighborhoods. In Proceedings of the 32nd ACM SIGIR Conference (2009)
9. Loll, F., Pinkwart, N.: Using Collaborative Filtering Algorithms as eLearning Tools. In Proceedings of the 42nd International HICSS Conference, (2009)
10. Maletsky, E. M. et al. Harcourt Math, Indiana Edition, Chicago: Harcourt. (2004)
11. Pinkwart, N., Aleven, V., Ashley, K., Lynch, C.: Toward Legal Argument Instruction with Graph Grammars and Collaborative Filtering Techniques. In Proceedings of the 8th ITS Conference, p. 227-236 (2006)
12. Robinet, V., Bisson, G., Gordon, M. B., Lemaire, B.: Inducing High-Level Behaviors from Problem-Solving Traces Using Machine-Learning Tools. In IEEE Intelligent Systems, 22, 4, p. 22-30 (2007)
13. Tsymbal, A.: The problem of concept drift: definitions and related work. Technical Report, TCD-CS-2004-15, Computer Science Department, Trinity College Dublin, Ireland. (2004)
14. Walker, A., Recker, M., Lawless, K., Wiley, D.: Collaborative Information Filtering: a Review and An Educational Application. In Int. Journal of AI in Education, 14:1–26 (2004)