Table of Contents

Preface

This volume contains the papers presented at WeASeL 2018: Optimizing Human Learning – Workshop eliciting Adaptive Sequences for Learning held on June 12, 2018 in Montréal.

Each submission was reviewed by at least 3 program committee members. The committee decided to accept 3 papers. The program also includes 2 invited talks and 1 tutorial.

What should we learn next? In this current era where digital access to knowledge is cheap and user attention is expensive, a number of online applications have been developed for learning. These platforms collect a massive amount of data over various profiles, that can be used to improve learning experience: intelligent tutoring systems can infer what activities worked for different types of students in the past, and apply this knowledge to instruct new students. In order to learn effectively and efficiently, the experience should be adaptive: the sequence of activities should be tailored to the abilities and needs of each learner, in order to keep them stimulated and avoid boredom, confusion and dropout.

Educational research communities have proposed models that predict mistakes and dropout, in order to detect students that need further instruction. There is now a need to design online systems that continuously learn as data flows, and self-assess their strategies when interacting with new learners. These models have been already deployed in online commercial applications (ex. streaming, advertising, social networks) for optimizing interaction, click-through-rate, or profit. Can we use similar methods to enhance the performance of teaching in order to promote lifetime success?

We thank the workshop chairs, Nathalie Guin and Amruth Kumar.

May 24, 2018
Tokyo, Japan

Michal Valko, Fabrice Popineau and
Jill-Jênn Vie

Optimizing Human Language Learning

Masato Hagiwara

Duolingo, Pittsburgh, PA, USA
masato@duolingo.com
http://masatohagiwara.net

Abstract. Learning foreign languages has become an essential skill in the globalized economy - English alone is estimated to have a total of 1.5 billion learners worldwide. As computer-based language learning apps increase in popularity, they generate vast amounts of student learning/behavioral data, opening up entirely new possibilities to optimize human language learning on an unprecedented scale.
In the first part of this talk, we introduce results from our user behavioral analysis and performance prediction projects using the learner data from Duolingo to find out the key traits of successful language learners. Secondly, we review some of the recent development to maximize second language learning through optimizing spaced repetition. Finally, we present the task of second language acquisition modeling (SLAM), which is a task to predict errors made by second language learners based on their past performance, along with some of the key findings from the SLAM shared task we hosted recently.

Keywords: Language learning · Spaced repetition · Second language acquisition modeling.

Where's the Reward? A Review of Reinforcement Learning for Instructional Sequencing

Shayan Doroudi

Computer Science Department
Carnegie Mellon University, Pittsburgh, PA, USA
shayand@cs.cmu.edu
http://www.cs.cmu.edu/~shayand/

Abstract. Since the 1960s, researchers have been trying to optimize the sequencing of instructional activities using the tools of reinforcement learning (RL) and sequential decision making under uncertainty. Many researchers have realized that reinforcement learning provides a natural framework for optimizing and personalizing instruction given a particular model of student learning, and excitement towards this area of research is as alive now as it was over fifty years ago. But does it actually help students learn? If so, when and where might we expect it to be most helpful? To help answer these questions, I will take three approaches. First, I will present a historical narrative of attempts to optimize instructional sequencing using RL. By looking to the past, we hope to better understand why researchers from different communities have worked on this problem and discover some trends that might tell us where the field is going. Second, I will present a case study of two experiments that we ran in a fractions intelligent tutoring system that showed no significant differences between various instructional policies. Finally, I will systematically review the empirical research in this area. We find that in many cases where RL has been applied to rich domains and environments, such as our intelligent tutoring system, it has not been very successful. However, I will show that it has been successful in settings that are constrained in one or more ways. Based on insights we draw from these three approaches, I make suggestions for how the field should proceed if we want to make the most out of reinforcement learning and if we want to quickly identify how rewarding this line of research might be. In particular, I suggest that data-driven RL approaches be informed by and constrained with ideas and theories from the learning sciences and that researchers perform more robust evaluations of instructional policies derived using reinforcement learning before testing them on students. I present on work conducted with Emma Brunskill and Vincent Aleven.

Keywords: Reinforcement learning · Sequential decision making · Student learning.

Knowledge Tracing Machines: towards an unification of DKT, IRT & PFA

Jill-Jênn Vie

RIKEN Center for Advanced Intelligence Project, Tokyo, Japan
vie@jill-jenn.net
https://jilljenn.github.io

Abstract. The goal of this tutorial is to make you compare typical baselines for predicting student performance (item response theory, performance factor analysis) on famous datasets, and replace some blocks of their architectures with deep neural networks (deep knowledge tracing, deep factorization machines). Hopefully we can understand where neural networks improve the predictions substantially, and where they do not. No knowledge of educational models is needed, an experience of Python is preferred. All code can be retrieved at https://github.com/jilljenn/ktm.

Keywords: Item response theory · Deep knowledge tracing · Predicting student performance.

SARLR: Self-adaptive Recommendation of Learning Resources

Liping Liu, Wenjun Wu and Jiankun Huang

State Key Lab of Software Development Environment Department of Computer Science and Engineering, Beihang University, Beijing, China
{liuliping,wwj,hjk}@nlsde.buaa.edu.cn

Abstract. Personalized recommendation is important for online students to select rich learning resources and make their own learning schedules. We propose SARLR, a new self-adaptive recommendation algorithm of online learning resources. The SARLR algorithm integrates an IRT-based learning cognitive model named T-BMIRT into the recommendation framework and is able to adaptively adjust learning path recommendations based on dynamic of individual learning process. The experimental results show that the SARLR algorithm outperforms the existing recommendation algorithms.

Keywords: Online Education, Learning Recommendation, ITS

1 Introduction

With the growing prevalence of online education, students have access to all kinds of electronic learning resources, including electronic books, exercises and learning videos. Given the diversity of students' background, learning styles and knowledge levels, it is essential to have personalized recommendation tools to facilitate students in choosing their own learning paths to satisfy their individual needs [1]. Previous studies have introduced personalized learning recommendation algorithms following the two major approaches including rule-based recommendation and data-driven recommendation.

Most Intelligent Tutor Systems (ITS) such as [2], primarily adopt the rule-based approach to design their recommendation algorithms, which requires domain experts to evaluate learning scenarios for different kinds of students and define extensive recommendation rules accordingly. Apparently, such a labor-intensive approach can only be applied in specific learning domains. For modern online educational systems, designers often take the data-driven approach by utilizing collaborative filtering methods to implement learning recommendation algorithms. These data-driven recommendation algorithms [3] attempt to identify suitable learning resources for students by comparing similarity among students and learning objects.

Although the data-driven recommendation approach is more scalable and general than the rule-based approach, current proposed solutions have common problems in achieving highly adaptive recommendation towards students' latent learning state. They often focus on either searching for similar learning resources based on content or identifying similar student groups based on their learning behaviors. The recommended learning objects or paths fail to consider the impact of difficulty of learning objects and dynamic change in students' learning states.

In this paper, we propose a novel learning recommendation algorithm named SARLR, which attempts to integrate an IRT-based learning cognitive model into the recommendation framework and to adaptively adjust learning path recommendations based on dynamics of individual learning process. Specifically, we introduce a temporal, multidimensional IRT-based model named as T-BMIRT, which can accurately infer student proficiency of multiple latent skills and difficulties of exercise assessments. In addition, the T-BMIRT model incorporates the parameter of video learning, which can describe the improvement in student skills after their interactions with video lectures. Based on the T-BMIRT model, the SARLR algorithm can comprehensively analyze every student's skill progress at each learning step and recommend to them a personalized learning path with the matching online video lectures and homework problems.

The contributions of this paper are the two-fold. First, we introduce the T-BMIRT model, to estimate students' latent skill levels and difficulties of learning resources for recommendation. Second, we propose the SARLR algorithm by integrating the T-BMIRT model in the adaptive recommendation process of learning resources. The experimental results confirm that the SARLR outperforms regular recommendation algorithms. Lastly, we present an evaluation strategy for recommendation algorithms in terms of rationality and effectiveness.

2 Related Work

Data-driven learning recommendation algorithms often utilize common recommendation methods widely adopted in the e-Commence area, including Collaborative Filtering (CF) and Latent Factor Model (LFM). CF can be further divided into UCF (User-based Collaborative Filtering) and ICF (Item-based Collaborative Filtering). The core idea of LFM is to connect users and items through latent features [4].

EduRank [5] is a collaborative filtering based method for personalization in e-learning. It can generate a difficulty ranking of questions for a target student by aggregating the ranking of similar students. Although this method is able to rank the available exercise questions based on their difficulties for similar students, it doesn't integrate cognitive learning models in its framework for estimating the ability of individual students. Thus, it can't generate the matching learning paths for students based on their state of latent skills.

The most related work to our research in previous studies is the Latent Skill Embedding (LSE) model [6], which also presents a probabilistic model of students and lessons. Although the LSE model provides a good foundation for designing a recommendation framework for personalized learning, the paper [6] doesn't propose a detailed recommendation algorithm. Our T-BMIRT model is more fine-grained than the LSE model because it defines a video learning parameter to capture student progress through their interaction with video lectures. Moreover, we present the SARLR algorithm that utilizes the T-BMIRT model to identify similar students for a target student and recommend their learning paths according to the dynamic state of the target student's latent skills. We also extend the recommendation evaluation criteria expected gain by incorporating two more metrics including relevance accuracy and difficulty accuracy. These new metrics can support more comprehensive performance evaluation for learning recommendation algorithms.

Recently, reinforcement learning has been explored in personalized study planning in ITS [7-9]. Most of them have not evaluated their approaches in real online learning scenarios and compared their performance to existing problem selection strategies used in current systems. Moreover, calculating an optimal personalized learning path in a POMPD is often time-consuming and even becomes intractable as the dimensions of the knowledge state and strategy spaces increase. Therefore, our SARLR algorithm adopts the collaborative filter based approach and we plan to investigate the possibility of utilizing reinforcement learning in our framework in future work.

3 SELF-ADAPTIVE RECOMMENDATION

Fig.1 illustrates the major components in the SARLR algorithm. First, it uses the T-BMIRT model to estimate every student's skill levels and difficulties of learning resources. Second, it searches for similar students based on their skill vectors from the outputs of the T-BMIRT model. Third, it extracts the learning path of the best student, whose skill level is the highest among the similar students after learning related knowledge. Lastly, it recommends the learning path to the target student and sets up two pre-warning conditions to adaptively adjust his recommended contents. The target student's latest behavior data are collected instantly and used as a feedback to update the T-BMIRT model. Thus, all of the modules form a closed loop, which constantly optimizes our model.

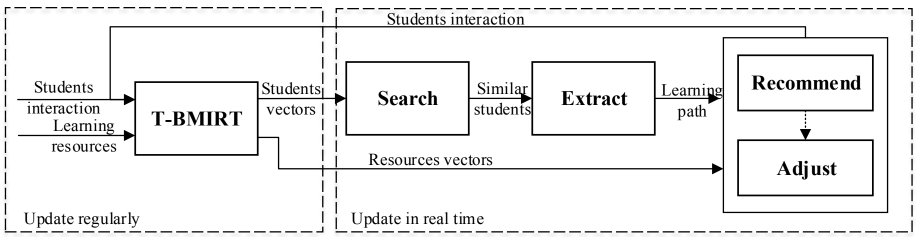

Fig. 1. The Overall architecture of the SARLR algorithm

3.1 The T-BMIRT model

The T-BMIRT model aims to model students and learning resources to infer students' latent skills and learning resources' attributes on multiple knowledge components. We define the model based on IRT, T-IRT and MIRT model [10]. In a two-parameter IRT model, the probability of the student s correctly answering the question q is given by:

$$p_{sq} = \frac{1}{1+exp[-(\alpha_q(\theta_s-\beta_q))]},\ P(\theta_{t+\tau}|\theta_t) = \phi_{\theta_t,\upsilon^2\tau}(\theta_{t+\tau}) \tag{1}$$

Where α_q is the question discrimination, β_q is the question difficulty, θ_s is the student's ability value. The Temporal IRT (T-IRT) model [11] extends the original IRT and MIRT model by modeling a student's latent skills over time as a Wiener process, where $\theta_{t+\tau} - \theta_t \sim N(\theta_t, v_2\tau)$. The model indicates the ability value of the student at the next moment is only relevant to his current ability value.

The T-IRT model only considers interactions between students and assessments, ignoring their interactions with learning videos. However, we believe that the students' ability can be significantly improved after completing a learning video. Therefore, in [12], we introduce a new model T-BMIRT by incorporating learning video parameters to describe the impact of students' interaction with learning videos. The major equations are defined in Eq (2):

$$P(\vec{\theta}_{s,t+\tau}|\vec{\theta}_{s,t}, \vec{l}_{s,t}) = \phi_{\vec{\theta}_{s,t}+\vec{l}_{s,t},\upsilon^2\tau}(\vec{\theta}_{s,t+\tau}),\ \vec{l}_{s,t} = \frac{d_{s_t}}{d_t}\cdot\vec{g}_t\cdot\frac{1}{1+exp\left(-\left(\frac{\vec{\theta}_{s,t}\cdot\vec{h}_t}{\|\vec{h}_t\|}-\|\vec{h}_t\|\right)\right)} \tag{2}$$

Where $\vec{l}_{s,t}$ represents knowledge that student s gains from the video t, $\vec{g}_t$ represents knowledge of the video t, $\vec{h}_t$ is the prerequisites of video t, d_{s_t} is the duration in which student s watches video t and d_t is the total length of the video t. In Eq (2), both student ability and learning video requirements have been expanded from one-dimensional to multidimensional. We utilize the vector projection method to determine whether the relevant abilities of the student exceed the relevant skill requirements of the video lectures.

The T-BMIRT model enables us to infer every student's current ability θ, video knowledge g and video skill requirements h through the student's responses of assessment questions. The detailed model fitting process of the T-BMIRT can be found in [12]. An approximation technique makes it possible to train the T-BMIRT in an online way. As a result, the T-BMIRT can be effectively used in the framework of the SARLR algorithm to estimate the parameters of learning resources and students' ability levels.

3.2 Similar Students Search and Learning Path Extraction

SARLR Phase 1 describes the process of searching similar students and extracting a suitable learning path for a target student. At Step 1, the algorithm identifies the students *MS* with the similar skill levels to the target student s_X through k-nearest neighbor search method over the k-dimension tree (kd-tree) structure and k-nearest neighbor search method. At Step 2-4, the algorithm selects the best student $s_b \in MS$ with the highest ability level at the moment when they complete learning specific knowledge units. At Step 5, the algorithm extracts the learning path p of s_b to the target student s_X.

SARLR Phase 1: Search and Extraction

INPUT:

Set of students $S = \{s_1, s_2, \dots, s_n\}$, target student $s_X \in S$
Matrix of abilities $A = [\theta_{s,t}]$, where $\theta_{s,t}$ is the ability value of student s at time t
Set of learning resources $E = \{e_1, e_2, \dots, e_m\}$
The time in this paper is the index of learning resources with the student just completed learning.

OUTPUT: learning path p

```
search for similar students MS, where s_k ∈ MS and θ_{s_k,t_0} is similar to θ_{s_X,t_0}
for each s_i ∈ MS do
    find s_b = argmax(distance(θ_{s_i,T_{s_i}} − θ_{s_i,t_0})), where T_{s_i} is the time of s_i completing learning
end for
extract the learning path p = (e_{i_1}, e_{i_2}, ... e_{i_T}) of s_b
return p
```

3.3 Adaptive Adjustment

Because each individual student has his/her inherent learning style, even when he follows the recommended learning path generated in SARLR phase 1, the learning outcome may not be as good as expected by the recommendation algorithm. In order to deal with this problem, we set up the two conditions in Eq (3) to initiate the Adaptive Re-planning phase, which is defined in *SARLR Phase 2*.

$$p_{sq} = \frac{1}{1+exp\left(-(\vec{\theta}_{s,i} \cdot \vec{\alpha}_q - b_q)\right)},\ p_{se} = \frac{1}{1+exp\left(-\left(\frac{\vec{\theta}_{s,i} \cdot \vec{h}_e}{\|\vec{h}_e\|} - \|\vec{h}_e\|\right)\right)} \tag{3}$$

Eq (3) specifies p_{sq} and p_{sl} to evaluate the progress of the target student in the learning path. p_{sq} indicates the probability of student s correctly answering exercise q, where $\vec{\theta}_{s,i}$, $\vec{\alpha}_q$ and b_q represent the same symbols as the T-BMIRT model in Eq (1-2). p_{se} indicates the degree of knowledge that student s can acquire from the video e, where $\vec{q}_e$ represents the level of knowledge required for the learning video.

When p_{sq} becomes less than the threshold C_{sq}, it means that the difficulty of the exercise q in the recommended learning path has significantly exceeded the student's ability. When p_{se} becomes less than the threshold C_{se}, it means that the skill level of the target student is lower than the requirement of the recommend video e , thus he can only acquire little knowledge from the video. When either condition is met, the SARLR determines that the original recommended path has to be re-planned to match the student's knowledge state.

SARLR Phase 2: Adaptive Re-planning

INPUT:

Target student s_X, recommended learning path $p = (e_{i_1}, e_{i_2}, \dots e_{i_T})$
Result of s_X interacted with learning resources in p

OUTPUT: new learning path

```
for each e ∈ p do
    if e is a video and p_se < C_se do
        return SARLR Phase 1 to re-plan path p
    else if e is an exercise and s_X failed it and p_sq < C_sq do
        return SARLR Phase 1 to re-plan path p
    end if
end for
```

4 EXPERIMENTS

We selected two datasets to perform our experiments, the public “Assistments”, including 224,076 interactions, 860 students, 1,427 assessments and 106 skills, and a blended learning data from our learning analysis platform including 14,037,146 learning behavior data from 140 schools and 9 online educational companies.

4.1 Experiments for T-BMIRT

We divided each data set into two parts, one part only contains single skill assessments, and the other part contains multiple skills assessments. The IRT, T-IRT are single skill models, and the MIRT and T-BMIRT are multiple skills models. The dimensions for models are related to the numbers of knowledge components. The values in Table 1 are average results of the cross-validation. It shows that T-BMIRT outperforms the other models on each dataset, especially on the multidimensional dataset.

Table 1. Prediction Results of each model

Models	Assistments				Blended learning data			
	One-dimensional		Multidimensional		One-dimensional		Multidimensional	
	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC
Frequency method	0.694	N/A	0.683	N/A	0.702	N/A	0.688	N/A
IRT	0.716	0.779	0.701	0.758	0.721	0.784	0.706	0.752
MIRT	0.714	0.771	0.721	0.786	0.718	0.775	0.722	0.783
T-IRT	0.738	0.805	0.712	0.769	0.744	0.801	0.717	0.764
T-BMIRT	0.743	0.815	0.738	0.803	0.757	0.820	0.748	0.816

4.2 Rationality Evaluation

The rationality evaluation verifies whether the algorithm can recommend the suitable learning resources that meet the student’s needs and ability levels. We set the following two indicators for it.

$$\mathrm{RC}_{\mathrm{s_x}} = \frac{\sum_{e_i}^{p} similarity(h_{e_i}, KC_{s_x})}{m},\ \mathrm{DC}_{\mathrm{s_x}} = \frac{\sum_{e_i}^{p} similarity(h_{e_i}, \theta_{s_{x,i}})}{m} \quad (4)$$

Where $e_i \in p$ is the learning resources in a recommended path, m is the length of the path, KC_{s_x} is the knowledge components which s_x is learning in the current chapter, function *similarity()* calculates the adjusted cosine similarity of the two vectors in the parentheses. The relevance accuracy RC_{s_x} is used to evaluate whether the difficulties of the recommended learning resources for the target student s_x are matched with his ability. The difficulty accuracy DC_{s_x} is set to evaluate whether the difficulties of the recommended learning resources for the target student can match his current ability levels.

We selected the blending data to do this experiments. Table 2 shows the average of the 10-fold cross-validation results. It can be seen that the UCF and ICF have a similar effect, but the UCF works better on the relevance accuracy, while the ICF is better at the difficulty accuracy. The LFM performs better than the first two algorithms in terms of both indicators. The SARLR algorithm performs best among all these algorithms.

Table 2. Results of Rationality Experiment.

Model	Relevance accuracy	Difficulty accuracy
UCF	0.86	0.77
ICF	0.71	0.83
LFM	0.87	0.84
SARLR	0.97	0.92

4.3 Effectiveness Evaluation

The effectiveness evaluation verifies whether the students' abilities can be improved by the recommendation algorithm. We clustered the students into six groups according their ability levels. We calculated "expected gain" $G = \frac{E\left(R_{S'}\right) - E(R_S)}{E(R_S)}$ by using PCA and K-means method to further split the students of the same group into two parts based on their learning paths [6]. One part is the students whose learning paths are strictly recommended, denoted as S' , and the other part is the students whose learning path are randomly selected, denoted as S. $E(R_{S'})$ and $E(R_S)$ and indicate that the students' average score in the last online assessment. We sorted the six groups of the students ascendingly based on their ability levels: group 1 has the lowest skill level, group 2 has a higher skill level than group 1, and group 6 has the highest.

Table 3. Results of Effectiveness Experiment

Model	Expected gain					
	1	2	3	4	5	6
UCF	-0.04	-0.06	0.07	-0.03	0.08	0.01
ICF	0.05	0.04	-0.03	0.07	-0.02	0.05
LFM	0.04	0.12	0.09	0.10	0.03	-0.05
SARLR	0.11	0.27	0.24	0.23	0.17	0.06

We selected the public data "Assistments" to do this experiments. Table 3 shows that the SARLR algorithm performs much better than the other three algorithms. Especially for the students in group 2 to group 5, the SARLR algorithm helps them to achieve noticeable progress from the recommendation learning paths. It indicates that SARLR is more effective on improving learning gain of students with average ability levels.

5 CONCLUSIONS

We developed a self-adaptive recommendation algorithm of learning resources (SARLR) to personalize students' learning path. It contains the T-BMIRT, a temporal blended multidimensional IRT model, which performs well on the prediction task of multi-dimensional skills assessments, especially when the study process contains learning video interactions. Based on the T-BMIRT model, the SARLR algorithm adopts a reasonable recommendation strategy and establishes conditions to adaptively adjust recommendations towards the dynamic needs of the students. In addition, we extend the evaluation criteria for personalized learning recommendation in term of rationality and effectiveness. Experimental results prove that the SARLR algorithm outperforms the other recommendation algorithms based on CF and LFM.

References

1. Akbulut, Y., Cardak, C. S.: Adaptive educational hypermedia accommodating learning styles: A content analysis of publications from 2000 to 2011. Computers & Education. 58(2), 835-842 (2012).
2. Vesin, B., Ivanović, M., KlašNja-MilićEvić, A., Budimac, Z.: Protus 2.0: Ontology-based semantic recommendation in programming tutoring system. Expert Systems with Applications. 39(15), 12229-12246 (2012).
3. Wu, D., Lu, J., Zhang, G.: A fuzzy tree matching-based personalized e-learning recommender system. IEEE Transactions on Fuzzy Systems. 23(6), 2412-2426 (2015).
4. Jenatton, R., Roux, N. L., Bordes, A., Obozinski, G. R.: A latent factor model for highly multi-relational data. In: Proceedings of the 25th International Conference on Neural Information Processing Systems, pp. 3167-3175. ACM, California (2012).
5. Segal, A., Katzir, Z., Gal, K., Shani, G., Shapira, B.: Edurank: A collaborative filtering approach to personalization in e-learning. In: Proceedings of the 7th International Conference on Educational Data Mining, pp. 68–75. EDM, London (2014).
6. Reddy, S., Labutov, I., Joachims, T.: Latent skill embedding for personalized lesson sequence recommendation. *arXiv preprint arXiv:1602.07029* (2016).
7. Theocharous, G., Beckwith, R., Butko, N., Philipose, M.: Tractable POMDP Planning Algorithms for Optimal Teaching in SPAIS. In: Workshop on Plan Activity, and Intent Recognition (PAIR), IJCAI (2009).
8. Folsom-Kovarik, J. T., Sukthankar, G., Schatz, S. L., Nicholson, D. M.: Scalable POMDPs for Diagnosis and Planning in Intelligent Tutoring Systems. In: AAAI Fall Symposium: Proactive Assistant Agents. AAAI Press, Virginia (2010).
9. Brunskill, E., Russell, S.: Partially observable sequential decision making for problem selection in an intelligent tutoring system. In Educational Data Mining (EDM), pp. 327–328 (2011).
10. Reckase, M.: Multidimensional item response theory. Springer, New York (2009).
11. Ekanadham, C., Karklin, Y.: T-skirt: Online estimation of student proficiency in an adaptive learning system. arXiv preprint arXiv:1702.04282 (2017).
12. Huang, J.,Wu, W.: T-BMIRT: Estimating representations of student knowledge and educational components in online education. In: 2017 IEEE International Conference on Big Data, pp. 1301-1306. IEEE Press, Massachusetts (2017).

An Adaptive Tutor to Promote Learners' Skills Acquisition during Procedural Learning

Joanna Taoum, Anaïs Raison, Elisabetta Bevacqua and Ronan Querrec

Lab-STICC, UMR 6285, CNRS, ENIB, France
{taoum,raison,bevacqua,querrec}@enib.fr

Abstract. Our research work proposes an adaptive and embodied virtual tutor based on intelligent tutoring systems. The domain model is represented in our work by a virtual environment meta-model and the interface by an embodied conversational agent. Our main contribution concerns the tutor model, that is able to adapt the execution of a pedagogical scenario according to the learner's level of knowledge. To achieve such a goal, we rely on the inference of the learner's memory content.

Keywords: Adaptive Pedagogical Behavior · Virtual Environment · Learner's Memory · Pedagogical Scenario · Embodied Conversational Agent

1 Introduction

The work presented in this paper is applied to the domain of procedural learning in a virtual environment for industrial systems. According to Anderson [1], procedural learning is considered to be complex and this complexity requires the use of practice (repetition). In order to be able to manage the interaction between a tutor and a learner during these repetitions, we choose to describe this information using pedagogical scenarios. These scenarios define the activities that should be carried out by the tutor and the learner, their sequencing, as well as the pedagogical objectives that should be achieved.

However, these scenarios remain general. They can be effective at the beginning of learning (during the first repetitions), but not in the following repetitions. Considering that each learner evolves differently, during repetitions, it is important to adapt the execution of these pedagogical scenarios according to the learner's evolution.

The real-time adaptation of the pedagogical situation to a learner is one of the major objectives of Intelligent Tutoring Systems (ITSs). In order to adapt the situation to the learner, a fundamental goal of an ITS is to model the learner. In procedural learning domain, Corbett and Anderson [2] propose some general concepts to model the learner during the acquisition of procedural skills. These concepts are too theoretical to be applied to teaching procedures in industrial systems. As we are dealing with teaching human activities in industrial systems, the cognitive knowledge that our student model infers is related to memorization. Atkinson and Shiffrin [3] proposed a general theoretical framework which divides human memory into three structural components: sensory memory, working memory and long-term memory. To implement this general framework of memory, several ITSs have been built using the cognitive architecture ACT-R [4]. The goal of ACT-R is to simulate the realization of complex tasks by human beings. It is mainly designed around two concepts: declarative and procedural knowledge. Declarative knowledge is represented by a set of chunks and procedural knowledge by a set of production rules (*if-then* statements). In ACT-R, information processing of memory is a *Black Box*. It can be used to generate the tutor behavior but not to represent the knowledge flow in the learner model.

In this work, we propose a tutor behavior that adapts the execution of the pedagogical scenario according to the learner's inferred knowledge (see section 3.1). To represent such a knowledge, we propose a cognitive architecture based on ACT-R [4]. In section 2, we introduce MASCARET [5] that we use to represent the domain model and the pedagogical scenario. To realize pedagogical assistances in a human-like way, we propose an interface model based on a virtual environment and an Embodied Conversational Agent (ECA).

2 Domain and Interface Model

The domain model is formalized in our work by MASCARET, a virtual reality meta-model based on UML. It allows to describe and simulate technical systems and human activities in a virtual environment. The domain expert uses class diagrams to describe the different types of entities, their properties and the structure of the environment. Procedures are designed as predefined collaborative scenarios through UML activity diagrams, which represent plans of actions. It is the role of the interface model to recognize when the student executes these actions. Using a meta-model to formalize the domain model 1) allows domain experts to provide the knowledge themselves in the ITS, and 2) keeps domain data explicit during the simulation, thus they can serve agents as the knowledge base.

In MASCARET, pedagogy is considered as a specific domain model. Pedagogical scenarios are implemented through UML activity diagrams containing a sequence of actions. These actions can be either pedagogical actions, like explaining a resource, or domain actions, like manipulating an object. For the definition of pedagogical scenarios and actions, we rely [6]. In MASCARET five types of pedagogical actions are implemented:

1. Pedagogical actions on the virtual environment: highlighting an object, playing an animation.
2. Pedagogical actions on user's interactions: changing the viewpoint, locking the position, letting the student navigate.
3. Pedagogical actions on the structure of the system: describing the structure, displaying a documentation about an entity.
4. Pedagogical actions on the system dynamics: explaining the procedure's objectives, explaining an action.

5. Pedagogical actions on the pedagogical scenario: displaying a pedagogical resource, making an evaluation (e.g. a quiz).

These pedagogical actions are realized through the interface model, that is represented in our work by an ECA, using GRETA platform [7]. This ECA is able to select and perform multi-modal communicative and expressive behaviors in order to interact naturally with the user. In MASCARET, any entity which acts on the environment is considered as an *agent*. Particularly, the ECA and the human user are *embodied* agents. An embodied agent is able to recognize as well as perform basic actions, like:

1. Verbal communication (e.g. giving an information)
2. Non-verbal actions (e.g. facial expression) and actions on the environment (e.g. manipulating an object)
3. Navigation (e.g. observing)

These basic actions are used to implement the domain and pedagogical actions involved the pedagogical scenario. Through the interface model, the tutor is able to recognize the realization of each of these actions performed by the user to evaluate the evolution of the pedagogical scenario and to adapt it if necessary.

3 Adaptive Tutor Model

The tutor model uses the knowledge of the domain model and the actions done by the learner in order to choose pedagogical actions that will be realized through the interface model. More precisely, the tutor behavior takes into account the actions done (or inaction) by the student by recognizing them through the interface. The goal of our proposed tutor model is to adapt the execution of the pedagogical scenario according to the student model represented in our work by the student's memory.

In what follows, we first describe the student model that is used to decide which adaptation to perform and then how the tutor behavior detects the need for adaptation.

3.1 Student Model

We propose a reimplementation of the generic framework of memory proposed by Atkinson and Shiffrin [3] in the context of learning procedures. Our contributions to this framework consist in making explicit the *Black Box* by 1) formalizing the user's memory information, and 2) implementing the transformation of the stimuli into knowledge and the knowledge flow between the three components of the human memory. In our work, incoming stimuli from the virtual environment and the virtual tutor are restricted to those related to vision and hearing. Thus, the student can see 3D objects and hear instructions uttered by the tutor about activities to realize. Therefore, we encode data about objects and activities. To formalize the encoding of information, we rely on MASCARET.

Objects are considered in Mascaret as `Entity`. An `Entity` can be hierarchical, thus it can be composed of `Entity` and represented by a name, geometric properties (position, orientation and shape) and domain model properties (as a meta class `Class` attribute). As for activities, they are represented by the meta class `Activity`, they can also be hierarchical and composed of several `Activity`, `Role`, `Action` and `Flow` between actions and objects. Mascaret data formalism is hierarchical, which allows to instantiate the content of the memories according to the knowledge level of the learner.

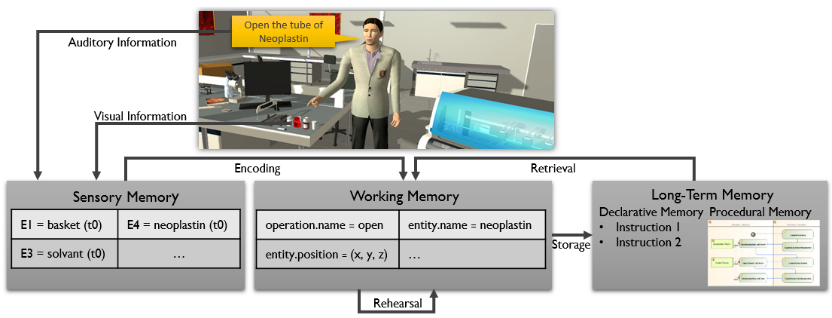

Fig. 1. Formalization of the encoding and structuring of instructions in the memory.

In this work, we therefore distinguish three structural components in human memory in which a sequence of cognitive processes is implemented to process information (encoding, storage, retrieval). The first operation involved in the information processing is the encoding of information. It is the transformation of incoming stimuli from the virtual environment and the virtual tutor to a formal representation that can be stored in the working memory. As mentioned previously, incoming stimuli are visual (set of objects in the student's field of view) and auditory (uttered by the tutor). Only prominent information (e.g. objects that have been highlighted by the tutor) is transferred from the sensory memory to the working memory. The working memory stores and manipulates information based on the content of the sensory memory and the long-term memory (prior knowledge). The level of complexity of stored information in the working memory depends on the student's prior knowledge (by complexity of information we mean the level of the formal representation in Mascaret hierarchical formalism). This prior knowledge is retrieved from the long-term memory. The transfer of some knowledge from the working memory to the long-term memory, takes place when the student completes an action [8].

This student model is used as an input in the tutor behavior.

3.2 Tutor Behavior

The tutor behavior takes into account the actions done by the learner and the inferred student model to adapt the execution of the pedagogical scenario. This adaptation can be a modification of the student model (modification of the memory content) and/or the execution of a pedagogical action. The decision making of the tutor behavior is represented in Figure 2.

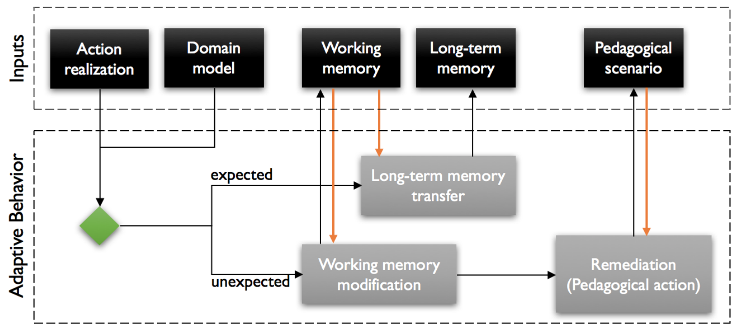

Fig. 2. Tutor behavior decision making.

The execution of a pedagogical scenario is a set of interaction between the tutor and the learner. As explained in section 2, the tutor actions (pedagogical actions) are realized through the interface, and this latter is also able to recognize the actions realized by the learner in the context of this interaction.

Our tutor behavior categorizes the actions done by the learner, based on two types of actions:

1. related to the domain model: an action can be either a domain action on a specific object or an answer to the tutor's questions. The tutor relies on the domain model to check if these actions are considered as errors or not.
2. related to the interaction: actions done by the learner can also be a feedback to the tutor's action (e.g. a facial expression, a question, observing the environment or an inaction). In this case, instead of using the domain knowledge, the tutor evaluates whether this feedback is negative or not.

If the learner's action is considered as an error or as a negative feedback, this means that this action is unexpected in the context of the executed scenario. In this case a new pedagogical action is needed and the content of the learner's memory must be reevaluated.

For example, if according to the pedagogical scenario the tutor explains the next action that the student has to do, we instantiate two chunks in the working memory, one for the `Action` and the other one for the `Entity`. If the student realizes an unexpected action (for example he/she shows a negative facial expression), then the tutor behavior considers that the student does not know the object position, contrary to what the tutor inferred. In this case the tutor remedies to this situation by re-evaluating the content of the student's working memory and then realizes a new pedagogical action to highlight the object.

4 Conclusion and Future Work

The model that we propose here allows an embodied conversational agent, playing the role of tutor, to execute a predefined pedagogical scenario written by a trainer in a virtual environment and especially to adapt its execution according to the individual evolution of students. To do this, the ECA infers the student's knowledge by estimating the content of his/her memories involved in procedural learning. The tutor behavior that we propose is a simple behavior that allows us to show the usability of the memory model that we have implemented to define a pedagogical behavior. In the same way that in Mascaret it is the trainer who describes the pedagogical scenario using a dedicated language (based on UML activities), we consider that it would be more interesting if it is the pedagogue which describes the tutor's behavior using the same language. We aim to make the concepts defined in our model accessible and formalized in this language.

In order to evaluate the impact of our model on the student's performance, we plan to carry an experiment that will involve two groups of participants. In the first group, a non-adaptive virtual tutor will be present in the virtual environment. The non-adaptive tutor will apply a single pedagogical scenario during repetitions. If the student asks for help, the tutor announces the action to be performed, its goal and highlights the object to manipulate. In the second group, an adaptive tutor will guide the learner. Based on our model, the tutor will be able to adapt the execution of the pedagogical scenario according the evolution of the learner's level of expertise. In this experiment, we expect that learners interacting with an adaptive tutor perform the procedure without errors and without the need for help, earlier than those who are interacting with a non-adaptive tutor.

References

1. Anderson, J.: A spreading activation theory of memory. Journal of verbal learning and verbal behavior **22** (1983) 261–295
2. Corbett, A., Anderson, J.: Knowledge tracing: Modeling the acquisition of procedural knowledge. User Modeling and User-Adapted Interaction **4** (1995) 253–278
3. Atkinson, R., Shiffrin, R.M.: Human memory: A proposed system and its control processes. In K. W. Spence and J. T. Spence (Eds.), The Psychology of learning and motivation: Advances in research and theory (vol. 2). (1968) 89–105
4. Anderson, J.: The architecture of cognition. 2nd edn. Psychology Press (2013)
5. Chevaillier, P., Trinh, T., Barange, M., Devillers, F., Soler, J., De Loor, P., Querrec, R.: Semantic modelling of virtual environments using Mascaret. In: Proceedings of the 4th Workshop SEARIS, IEEE VR, Singapore. (2001)
6. Le Corre, F., Hoareau, C., Ganier, F., Buche, C., Querrec, R.: A pedagogical scenario language for virtual environment for learning based on uml meta-model. application to blood analysis instrument. In: International Conference CSEDU. (2014) 301–308
7. Niewiadomski, R., Bevacqua, E., Mancini, M., Pelachaud, C.: Greta: an interactive expressive eca system. In: 8th International Conference AAMAS. (2009) 1399–1400
8. Ganier, F.: Factors Affecting the Processing of Procedural Instructions: Implications for Document Design. IEEE Transactions on Professional Communication **47**(1) (2004) 15–26

Optimizing Recommendation in Collaborative E-Learning by Exploring DBpedia and Association Rules

Samia Beldjoudi[1], Hassina Seridi[2]

[1]Superior School of Industrial Technologies, Annaba, Algeria
[1, 2]Laboratory of Electronic Document Management LabGED Badji Mokhtar University, Annaba, Algeria
[1]s.beldjoudi@epst-annaba.dz [2]seridi@labged.net

Abstract. Social tagging activities allow users to add free annotations on resources to express user interests, preferences and automatically generate folksonomies. This paper demonstrates how structured content available through DBpedia can be leveraged to support recommendation of resources in folksonomies. A limitation of resources' recommendation is the content overspecialization conducting in the incapability to recommend relevant resources different from the ones that the learner already knows. To address this issue, we proposed to take advantage of the richness of the open and linked data graph of DBpedia and association rules to learn learners' behavior. The proposed approach demonstrates the efficiency of using DBpedia to enhance diversity and novelty when recommending resources to users in folksonomies. The basic idea is to iteratively explore the RDF data graph to produce novel and diverse relevant recommendations.

Keywords: Collaborative E-learning, Recommendation, DBpedia, Diversity, Novelty.

1 Introduction

Social tagging systems have achieved a great success over the web in the last years, especially in recommendations approaches. The problem of a precise recommender system is that the entire set of recommended resources may be obvious as one considers the case of a film recommendation algorithm that only returns films of the same actor. To overcome this problem, novelty and diversity should be also considered in the evaluation of a recommender system, as precision only offers an incomplete description of the system's effectiveness.

The main focus of our study is how to exploit the semantic aspect of DBpedia to enhance resource recommendation within social tagging systems. We propose a new method for analyzing learner profiles according to their tagging activities in order to improve the recommendation of resources. The effectiveness of results depends on the resolution of social tagging drawbacks. In our process, we demonstrate how we can reduce the tags ambiguity problem by taking into account social similarities calculated on folksonomies combined with similarities between resources in DBpedia. We used also the force of Linked Open Data (LOD) to enhance resource recommendation by exploring the interlinked entities in LOD cloud. We base up on the iterative exploration of the DBpedia graph to obtain novel and diverse recommendations that should satisfy the learner and create the effect of surprise by recommending resources that the user did not expect at the beginning.

This paper is organized as follows: Section 2 is an overview of the main contributions related to our work. Section 3 is dedicated to the presentation of our approach. In section 4 we present and discuss the results of some experiments we conducted to measure the performance of our approach. Conclusion and future works are described in Section 5.

2 Related works

Social web based approaches, like folksonomies, have achieved a high level of improvement even in E-learning practice. In this section, an overview about some contributions attached to this field is proposed. [Kopeinik et al., 2017] investigated the application of two tag recommenders that are inspired by models of human memory. The authors find that displaying tags from other group members helps significantly in semantic stabilization in the group, as compared to a strategy where tags from the students' individual vocabularies are used. In [Beldjoudi et al., 2016], the authors proposed a new approach for personalizing and improving resources retrieval in collaborative learning with tackling tags ambiguity and event detection impact on resourced retrieved by ranking. In another contribution [Beldjoudi et al., 2017] proposed a method to analyze user profiles according to their tags in order to predict interesting personalized resources and recommend them. The authors proposed a new approach to reduce tag ambiguity and spelling variations in the recommendation process by increasing the weights associated to web resources according to social similarities. They base upon association rules for discovering interesting relationships among a large dataset on the web. [Karabadji et al., 2018] proposed to focus mainly on the growing of the large search space of users' profiles and to use an evolutionary multi-objective optimization-based recommendation system to pull up a group of profiles that maximizes both similarity with the active user and diversity between its members. In such manner, the recommendation system will provide high performances in terms of both accuracy and diversity. In our work we want to leverage the social and semantic web in order to enhance educational resources recommendation in collaborative e-learning.

3 Approach description

In this paper, we propose a method to analyze learner profiles according to their tags in order to predict interesting personalized resources and recommend them. We argue that the automatic sharing of resources strengthens social links among learners and we exploited this idea to reduce tag ambiguity in the recommendation process by increasing the weights associated to web resources according to social similarities. We based upon association rules that are a powerful method for discovering interesting relationships among a large dataset on the web. Our goal was to find correlations between tags, i.e. to find tags frequently appearing together, in order to extract those which are not used by one particular learner but which are often used by other users close to him in the social network.

The effectiveness of the recommendation depends on the resolution of the problems of folksonomies. In our approach we tackle the problems of tag ambiguity, diversity and novelty. To resolve the problem of tag ambiguity in recommendation, we propose to measure the similarity between learners to identify those who have similar preferences and therefore adapt the recommendation to learner profiles.

- First step: For each extracted association rule (Tags $A \rightarrow$ *Tags B*) whose antecedent applies to an active learner *lx*, we measure the similarities between this learner and the learners of his social network who use the tags occurring in the consequent of the rule. The resources associated to these tags are recommended to the learner depending on these similarities. To measure similarity between two learners (*l1* and *l2*), both are represented by a binary vector representing all their tags and we compute the cosines similarity between the two vectors.

- Second step: To avoid the cold-start problem which generally results from a lack of data required by the system in order to make a good recommendation, when the learner of the recommender system is not yet similar to other users, we propose to exploit semantic links between resources in DBpedia. DBpedia can be a reliable and rich source of content information that supports recommender systems to overcome problems, such as the cold-start problem and limited content analysis that restrict many of the existing systems, by building on a robust measurement of the similarities between resources using DBpedia. In this approach, we use the Linked Open Data to assess the similarity between folksonomies resources using their corresponding resources on DBpedia (i.e. we measure the similarity between the resources that would be recommended by the system, as related to a tag occurring in the consequent of an association rule, and those that are already recommended to the learner). The similarity between two resources is calculated using Jaccard index.

In another hand, when using a recommender system such as those of online stores, the results are mainly expected by the users. In this case, it is clear that the recommendation is not very helpful in the sense of the lack of diversity and novelty. To solve this dilemma in folksonomies-based collaborative learning, we propose extracting the most popular features found in the resources-based learner profile (i.e. the characteristics that interest the learner when they tag their resources) and then explore the LOD to extract resource linked with these features.

Let us consider a learner profile composed from the resources (*R1, R2, R3 and R4*), Thus the intersection between the resources' features must be calculated ($R1 \cap R2 \cap R3 \cap R4$), this is done because we want to extract the most popular characteristics that interest the learner when they choose tagging their resources. Then for each feature (*Pi*) in the result of intersection we will explore the LOD graph in the first level to extract other resources (*R5*) having these features or having a direct/ indirect link with these later (*R6, R7* resp).

Supposing that ($R1 \cap R2 \cap R3 \cap R4$) = {[domain: informatics]; [author: …]; [year: …]; [edition: …]…}. By exploring the LOD graph we find that the resource "informatics" is linked with other resources (for example: "University, Formation, Bio-Informatics…") via the predicates (*P1, P2, P3…*). In its turn the resources "University, Formation, Bio-Informatics…" are linked via other predicates (*Pj*) with other resources (for example: "Boston University…"). Therefore, it appears relevant to recommend some courses of the Boston University to the current user.

Our approach is based on the iterative exploration of the DBpedia graph, where each step depends on the result of the previous steps. In order to obtain relevant and personalized recommendations for each learner, we calculate the occurrence number of the {domain, author, year, edition…} characteristics and then we choose the ones that best reflect the learner interest to exploit them later in the exploration of the RDF graph of DBpedia.

The purpose of the graph exploration is to obtain recommendations that should not only satisfy the learner but also to have diversity and a novelty in the recommendation, to create the effect of surprise by recommending resources that the learner did not expect at the beginning. The learner evaluates the recommended resources in real time in each iteration. The process stops when none of the recommended resources has satisfied the user.

If the learner liked at least one resource among those in the proposed list, in the second iteration, we focus on these ones. Thus, we re-explore the LOD graph again starting from these items by using the query language SPARQL to return more educational resources connected with them; this technique allows us to propose a list of diverse and novel resources to ensure the surprise effect.

The real-time evaluation process as well as the exploration of the graph is iterative. At each iteration, we explore the graph based on the positive ratings assigned to the resources previously recommended. Indeed, the evaluation is an essential step to determine the new pattern of requests for the re-exploration of the graph to generate another list of recommendations. At each step, we propose to the user 10 resources, if he assigns a rating more or equal to three, we consider that he liked the recommended resource, and so we record it in his profile, otherwise we move to another resource.

After evaluating the 10 resources, the program suggests to the user to recommend after the 10 resources have been evaluated, the program suggests to recommend some more to the user. If he accepts then another list of resources is generated from his profile, otherwise, we stop and return the list of resources liked. With this method, we ensure that the recommended list of resources is diverse, where every user can obtain diverse resources even if they do not appear in the profile of his neighbors in the social network.

4 Experimental Results

In this section, experiment over a popular dataset is described and results are analyzed and discussed. The dataset exploited in our test is del.icio.us. In this experiment, we were interested in data generated from users who tagged resources about education. Thus, our database comprises 1128 tag assignments involving 95 users, 432 tags containing ambiguous tags and 314 resources.

4.1 Experimental Methodology

To evaluate the quality of a recommender system, we must demonstrate that the recommended resources are really being accepted and added by the users. Because the knowledge of this information requires asking the users of the selected databases if they appreciated the proposed set of resources, which is impossible in our case because we do not have access to this community, we have used a cross-validation where we have randomly removed some resources from the profile of each user, and we applied our approach on the remainder dataset in order to show if it can recommend the removed resources to their corresponding users or not. If it is the case, so we can conclude that our approach enables to extract the user preferences.

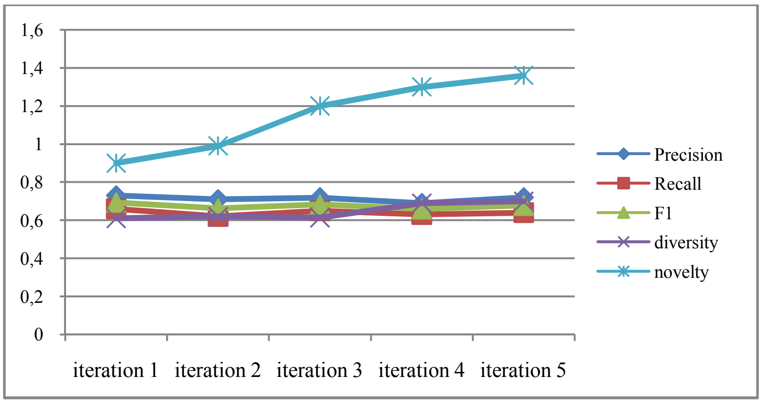

Fig. 1. Average precision, recall, F1, diversity and novelty of the recommendations

The curve presented in figure 1 show average values of precision, recall, F1, diversity and novelty measures in the five iterations. We notice that the precision achieved a good value in all iterations, this is due to the fact that the system recommends exactly the items wanted by the user i.e. those that match his profile. Sometimes the system begins to deteriorate in terms of precision but always with a value that exceeds 0.6. This decrease is quite normal since the system begins to recommend items according to different attributes (domain, year ...) which is known as diversity of recommendation. Learners sometimes accept the recommended resources and other times it was not the case. Recall and F1 measure achieved all both good values in the all iterations.

To calculate individual diversity and novelty, we used the metrics proposed in [Zhang and Hurley, 2009] and [Vargas, 2014] respectively. Figure 1 showed promising values of both diversity and novelty in the five iterations. This demonstrates the importance of DBpedia to extract more diversified and novel resources in the recommendation. It is clear that the effectiveness of recommendation depends of preserving both precision and diversity. Results demonstrate that our approach preserving both them in all iterations.

5 Conclusion

In this contribution we have exploited the strength of social aspect in folksonomies to let members in the community benefit from the educational resources tagged by other users, based on the recommendation of resources. The proposed approach is based on DBpedia, the objective was to overcome the problem of diversity and novelty in recommendation. Primary results show also the utility of exploring LOD graph in ensuring diversity when recommending personalized educational resources in social tagging systems. In order to continue and improve our work, we aim at using others principles like event detection, for example, to help capturing and analyzing the behavior of learners when new events come, this can improve recommendation and even resources ranking.

References

1. Beldjoudi, S., Seridi, H. & Benzine, A.: The Impact of Social Similarities and Event Detection on Ranking Retrieved Resources in Collaborative E-Learning Systems. In: Koch F., Koster, A., Primo, T., Guttmann, C. (eds) Advances in Social Computing and Digital Education. CARE 2016, SOCIALEDU 2016. Communications in Computer and Information Science, vol 677. Springer, Cham (2016)
2. Beldjoudi, S., Seridi, H. & Faron-Zucker, C.: Personalizing and Improving Resource Recommendation by Analyzing Users Preferences in Social Tagging Activities. Computing and Informatics 36(1): 223-256 (2017)
3. Karabadji, NEI., Beldjoudi, S., Seridi, H., Aridhi, S. & Dhifli, W.: Improving memory-based user collaborative filtering with evolutionary multi-objective optimization. Expert Syst. Appl. 98: 153-165 (2018)
4. Kopeinik, S., Lex, E., Seitlinger, P., Albert, D. & Ley, T.: Supporting collaborative learning with tag recommendations: a real-world study in an inquiry-based classroom project. In: LAK '17 Proceedings of the Seventh International Learning Analytics & Knowledge Conference Pages 409-418 (2017).
5. Vargas, S.: Novelty and diversity enhancement and evaluation in recommender systems and information retrieval. In Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR ’14, pages 1281–1281, New York, NY, USA, (2014).
6. Zhang, M. and Hurley, N.: Novel item recommendation by user profile partitioning. In 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology, volume 1, pages 508–515, Sept (2009).