US-China Education Review, ISSN 1548-6613
April 2011, Vol. 8, No. 4, 482-491

A Novel Approach for Enhancing Lifelong Learning Systems by Using Hybrid Recommender System

Ahmad A. Kardan, Omid R. B. Speily, Somayyeh Modaberi
Amirkabir University of Technology, Advanced E-Learning Technologies Laboratory, Tehran, Iran

The majority of current web-based learning systems are closed learning environments where courses and learning materials are fixed, and the only dynamic aspect is the organization of the material that can be adapted to allow a relatively individualized learning environment. In this paper, we propose an evolving web-based learning system which can adapt itself to its learners. More specifically, the novelty with respect to the system lies in its ability to find relevant content on the web and its ability to personalize and adapt this content based on the system's observation of its learners and the accumulated ratings given by the learners. Hence, although learners do not have direct interaction with the open web, the system can retrieve relevant information related to them and their situated learning characteristics. Lifelong learning scenarios have particular differences in their need for personalized recommendations that make not possible reusing existing general approaches of recommender systems. The paper describes those challenges and presents a hybrid proposal that combines different recommendation techniques to navigate the learner in a learning process and make lifelong learning systems personalized.

Keywords: lifelong learning, recommender systems, personalization

Introduction

Research on e-learning has gained more and more attention thanks to the recent explosive use of the Internet. The LLL (lifelong learning) paradigm supports the idea that learning should occur throughout a person's lifetime (Santos & Boticario, 2008; Taghipour, 2004). This paradigm promotes a learner-centered approach that removes social, physical and cognitive barriers, where dynamic support may foster attitudes and skills to improve the effectiveness of the learning process. In mediating this process, technology is playing an important role. In this sense, a dynamic support that recommends learners what to do to achieve their learning goals is desirable. Traditionally, ITS (intelligent tutoring systems) find mismatches between the knowledge of the expert and learner behavior or action that shows internalization of the knowledge by the learner. Therefore, these systems use cognitive methods and analysis of learners' actions to predict their knowledge and proposed instruction, thus to improve knowledge (Santos & Boticario, 2008). But ITS are domain-specific and limited to the domain which they have been designed. Another limitation of these systems is that the expert module,

---

Ahmad A. Kardan, Ph.D., assistant professor, Department of Computer Engineering and Information Technology, Amirkabir University of Technology, Advanced E-Learning Technologies Laboratory.
Omid R. B. Speily, Department of Computer Engineering and Information Technology, Amirkabir University of Technology, Advanced E-Learning Technologies Laboratory.
Somayyeh Modaberi, Department of Computer Engineering and Information Technology, Amirkabir University of Technology, Advanced E-Learning Technologies Laboratory.

which is embedded in ITS, cannot address all the possible responses to cover specific needs of each student at all learning process.

However, the majority of current web-based learning systems are static learning environments, where courses and materials are fixed, and the only dynamic aspect is the organization of the material that can be adapted to allow a relatively individualized learning environment. In this paper, because of the LLL nature, we use open LMS (learning management system) that has better usage in LLL, otherwise, systems will confront with many problems such as freshness, updating of system and inefficient content recommender. We will propose a multitasking open web-based learning system which can adapt itself not only to its learners, but also to the open web in response to the usage of its learning materials. Our system is open, in the sense that learning items related to the course could be added, adapted or deleted. Open LMS provide adding, updating or deleting learning contents that related to the course via world web. Our proposed e-learning system adapts both to learners and the open web by mediating between learners and the open web. In addition of using fixed materials, learner requirements process and system to create query for new materials related to requirements. In a traditional adaptive e-learning system, the delivery of learning material is personalized according to the learner model. However, the materials inside the system are fixed and determined by the system designer. In open lifelong system, learning materials are automatically found on the web and integrated into the system based on learners' interactions with the system. In this paper, we apply some filtering techniques that eliminate unsuitable materials that found on the web. Although learners do not have direct interaction with the open web, new or different learning materials in the open web can enrich their learning experiences (TANG & Mccalla, 2007). Other ability of our systems is working powerful in critical fields and high tolerance in unknown situation like new generation of science with related information shortage or new learner with no specification of his interests. Another superiority of our systems is suitable architecture for social networks like facebook. There are some similarities between social networks and lifelong learning, therefore, we think social networks can be used in learning. We propose combination of different adapted recommendation algorithms to address lifelong systems requirements. In this approach, the LLL system should be domain-specific, for example, "information technology LLL system" or can be developed as the social education network. Users can join the system according to their requirements and interests, and then the system guarantees that its user can get all new content recommendations based on their characteristics.

The organization of the paper is as follows: In section 2, we overview the related work done in recommender systems in LLL, focusing more on recent systems. We introduce our solution including high level architecture and required details in section 3. The conclusion of the paper comes in section 4 along with some recommendations for future work.

Related Work

Work of LLL systems is in the initial stage, but it is improving quickly. Santos and Boticario (2008) introduced inclusive scenarios of recommender systems and LLL and proposed recommending strategies for LLL. Drachesler, Hummel, and Koper (2008) proposed a combination of memory-based recommendation techniques that appear suitable to realize personalized recommendation on learning activities in context of e-learning. As described earlier, our proposed e-learning system makes individualized recommendations of materials for learners chosen from a dynamically evolving paper repository. There are several related works concerning tracking and recommending technical papers. Basu, Hirsh, and Cohen (2001) defined the paper recommendation problem as: "Given a representation of my interests, find me relevant papers" (pp. 231-252). They studied this issue in the context of assigning conference paper submissions to review committee members. Reviewers do not need to key in their research interests as they usually do, instead, a novel autonomous procedure is incorporated in order to collect reviewer-interested information from the web. Bollacker, Lawrence and Giles (1999) refined CiteSeer, through an automatic personalized paper tracking module which retrieves each learner's interests from well-maintained heterogeneous learners' profiles. Woodruff, Gossweiler, Pitkow, Chi, and Card (2000) discussed an enhanced digital book with a spreading-activation mechanism to make customized recommendations for readers with different types of background and knowledge. McNee et al. (2002) investigated the adoption of collaborative filtering techniques to recommend papers for researchers. They did not address the issue of how to recommend a research paper, but rather, how to recommend "additional" references for a target research paper. In the context of an e-learning system, additional readings in an area cannot be recommended purely through an analysis of the citation matrix of the target paper, because the system should not only recommend papers according to learners' interests, but also pick up those not-so-interesting-yet pedagogically suitable papers for them (McNee et al., 2002). In some cases, pedagogically valuable papers might not normally be of interest to learners, and papers with significant influence on the research community might not be pedagogically suitable for learners. Therefore, we cannot simply present all highly relevant papers to learners, instead, a significantly modified recommending mechanism is needed (TANG & Mccalla, 2007; Iglesias, Martinez, Aler, & Fernandez, 2009).

Proposed Approach

E-learning introduces another learning channel that services without any restrictions on time and space, and engages in distance-based, non-synchronized learning activities. However, in most e-learning platforms, convenience offered by digital knowledge content is more emphasized than integrating suitable learning theory into the e-learning, neglecting learning theory and practice. Consequently, these platforms do not have the ability to enable the learners to solve problems and are limited to provide services of teaching materials management but not sufficient practical knowledge required for solving students' learning problems. Although e-learning easily provides learning resources, without taking into account the characteristics of problems being encountered, the large amount of learning resources will result in cognitive overload or disorientation. Problem-solving is knowledge intensive. It involves three activities, including acquiring relevant knowledge to identify the core causes of a problem, developing solutions and taking appropriate actions to solve the problem (Iglesias, Martinez, Aler, & Fernandez, 2009). According to these activities, this paper introduces problem based on the architecture which indicated in Figure 1.

This paper proposes two stages of processing: (1) online; and (2) offline. In offline stage, contents should be organized in the conceptual clusters according to the concept that they explain. In order to make our solution both general and applicable, we chose to exploit hierarchical and conceptual document clustering which can provide us with semantic relationships between pages without the need of a specifically devised ontology—concept hierarchy or manual assignment of concepts to pages. An important factor in our selection is the ability of the method to perform incremental document clustering, since we prefer to come up with a solution that is able to cope with the changes in the web site content and structure and solve "the new item" problem that familiar with many of recommendation algorithms. To map pages to higher level concepts, we used the WebDCC (Web Document Conceptual Clustering) (Taghipour, 2004; Bobadilla, Serradilla, Hernando, &

MovieLens, 2009) clustering algorithm on the web pages. It is an incremental hierarchical clustering algorithm, which is originally invented to deduce users' needs, and falls in the category of conceptual clustering algorithms as it assigns tags to each cluster of documents. We apply this technique to organize our documents similar to the manner in which they are assigned to nodes of a concept hierarchy (Taghipour, 2004). We can do all these activities in offline stage. In online stage, we propose conversional method between learners and system.

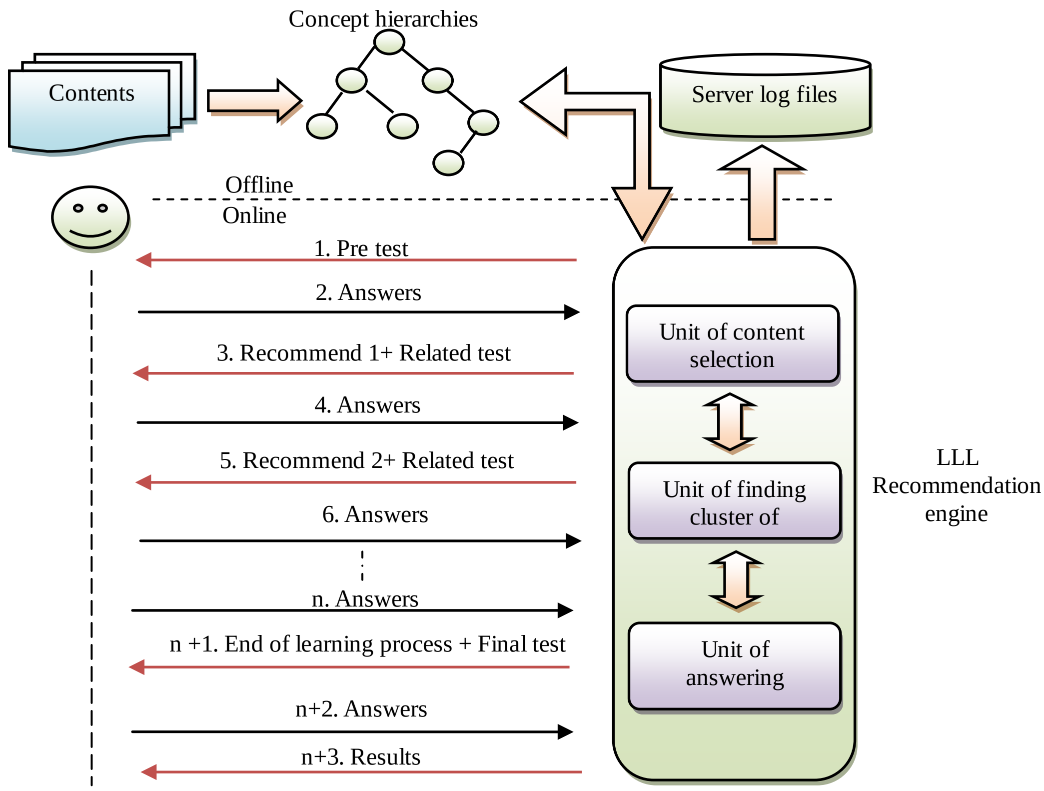

Figure 1. Proposal for combining recommending techniques in LLL.

Learners that join to this system regularly receive contents from LLL systems. At every specified time (for example, once in a month), system sends a questionnaire to its members. When a member or learner receives that questionnaire, they send them back to system after answering. By analysis of the learners' answers, system can recognize learners' needs and which part of learners' information should be updated. This analysis assigns to the unit of answering analysis. After the capturing of the learners' needs, systems should map needs to the related cluster of concept in concept hierarchies. Unit of content selection selects suitable content from whole cluster that recognized in the unit of "finding cluster of concept" as discussed before. Suppose that there are several clusters, such as "electronic commerce" in "information technology" LLL system, and this cluster includes different types of contents (journal papers, conference papers and technical report), publishing date and quality levels, we need the method to select suitable content between various types of contents in the cluster. For selecting the suitable content, we introduce hybrid solution that is specified particularly for LLL systems which is discussed in next section. This paper extends our previous method that discussed in (Ahmad, Kardan, Omid, Speily, & Somayye, 2009).

LLL Recommendation Engine

We describe our system in four phases (see Figure 2): (1) input; (2) process; (3) output; and (4) feedback processing. Input phase includes three types of inputs: (1) Candidate items are contents that recommender systems select “n” number of them for recommendation; (2) Actors that include four types of roles; and (3) The input information, such as learner models, friend weights, learning map, and so on. In process phase to make recommendation, all inputs are processed. Recommended items are presented to the learner, and his/her feedbacks are collected in the output phase. Finally, by processing feedbacks, system can update itself to predict and recommend better. Feedback processing phase provides restoration by reforming learner modeling, friends’ weight and other related essential information to increase system accuracy. Our proposed approach summarized in Figure 3 with more details.

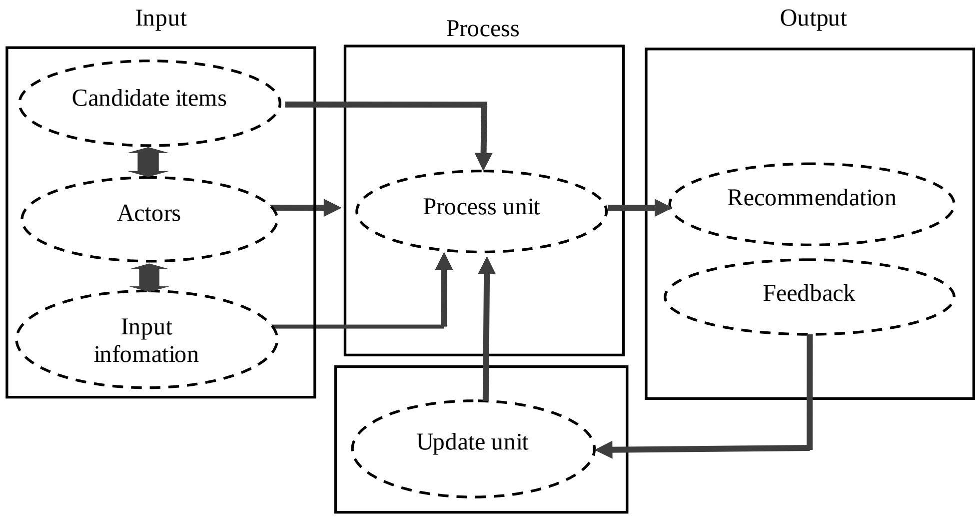

Figure 2. Concept model of our proposal.

Input phase. In this phase, actors have four roles, including learner, friends, group member and teacher. Friends are learners that directly interact with targeted learner. Interests and opinions of his/her friends, according to their similarity to targeted learner, have different weights. These weights are applied in producing recommendation. Another role is group member that indirectly interacts with targeted learner, and system uses them to give more accurate recommendations. If a group member’s interests and opinions are similar to targeted learner, the system will recommend him/her to add this member as a friend. By increasing number of friends and updating their weight, a better clustering is made, and consequently, system accuracy is increased. Also, this method works well when learner has few friends. Another role is teacher who has enough knowledge about the discussed topics in learning group. System can make a learning group without a teacher (a teacher can be an intelligent agent). This is a notable attribute of system, especially when learning group topic is very new or advanced, and an adequate teacher can not be found. Most important teacher works in this system listed as follows:

(1) Learning contents recommendations;

(2) Submission of recommendation when the system recommendation value is bellow 2 (Recommendation value is a parameter from 0 to 5 and calculates at the time of proposing it);

(3) Submission of learners’ annotation or summarization after they study learning content.

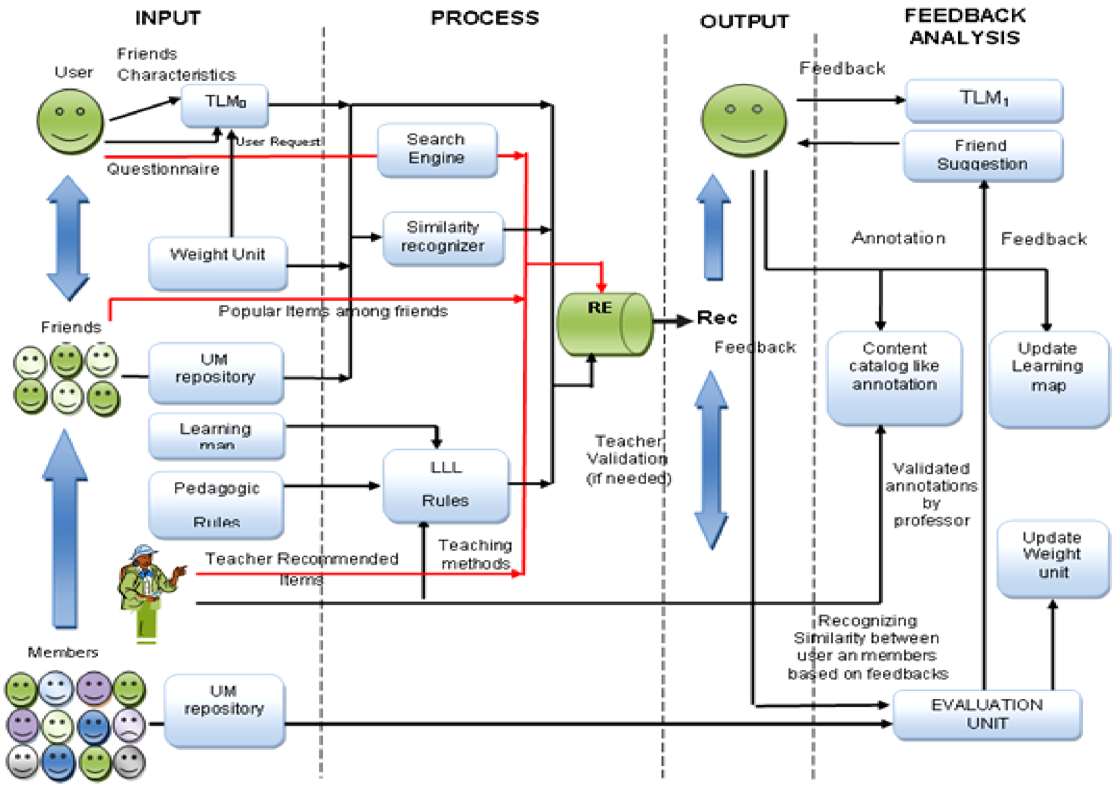

Figure 3. Proposal for combining recommending techniques in LLL.

One of the important elements in lifelong learning system is the learner modeling. Because of accuracy and efficiency of two partial learner modeling methods (Kardan & Einavypour, 2008), we use a modified version of it. Figure 4 shows an overall view of proposed learner modeling approach. At first, system does not have any idea about learner, therefore, to accomplish this problem, it uses questionnaire and inviter learner model. For joining learning group, each learner should have invitation from members of the learning group. Also, he/she can alternatively answer the questionnaire, including questions about a learner's individual information, such as age, geographical location, religion, educations, etc., as well as questions about the relation between the learner and members who invite him/her, such as how much he/she knows inviters, how he/she be familiar with them, etc.. TLM_0 (Temporary Learner Model) shows a learner's temporary model at first stage (Iglesias et al., 2009).

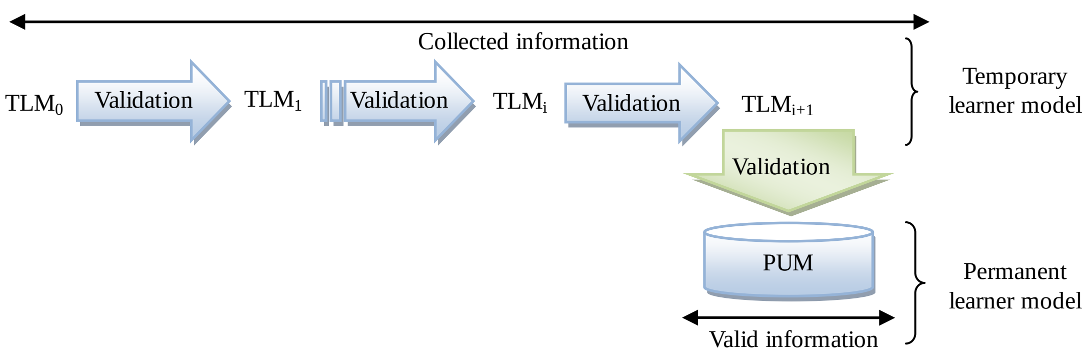

Figure 4. Learner modeling.

After a learner's interaction with system, system validates TLM_0 with considering learner's feedbacks. According to the differences between the learner model and real learner's activity, TLM_0 is updated to TLM_1 which is the learner's temporary model at the second stage. This process repeats "n" times, the value of "n" relates to system efficiency, and then TLM_n which is the learner's temporary model at the last stage and converts to the learner's permanent model.

We propose ranking and tagging paper based on paper publication time, paper level according to learner (beginner, average and expert) and method of teaching. Learning map has conceptual correlation with pedagogical rules. This map is saved for every learner and helps them to see their learning process. System by using this map, finds which content has been learned. Learning map can be visited by every friend to know new activities in the group (see also sample learning map in Figure 5).

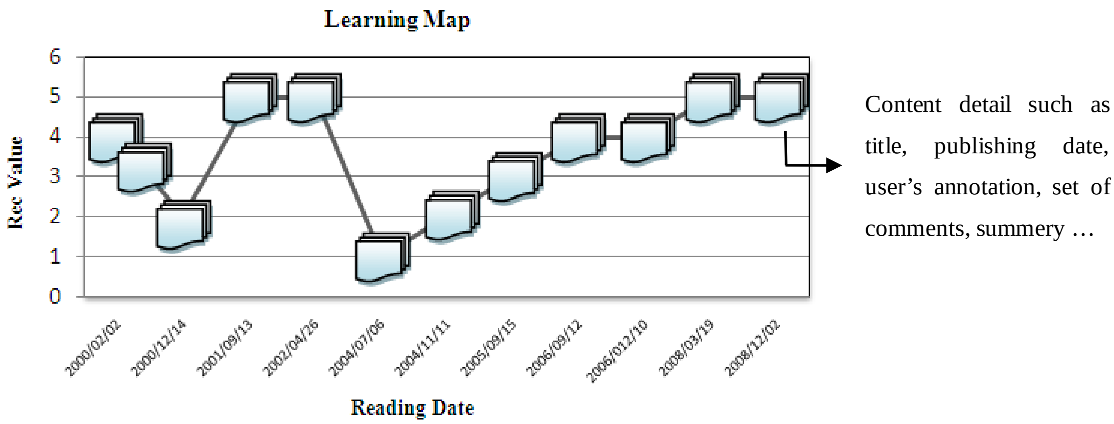

Figure 5. Learning map indicates records about user's activities.

Process phase. All processes and recommendation are done at this phase. We propose a mixture approach for making recommendation in lifelong learning systems. In contrast to common approaches that work with limited amount of content, our proposed approach let learners contact with universal web and search needed content through web at time of learning process. CF (collaborative filtering), as the most important methods in recommendation, is not proper approach for lifelong learning system, because the LLL nature is working with varied and very detailed information. So when system has information leakage about learners preferences, it cannot work well. This situation in LLL is more probable. To accomplish this weakness, we mix it by an efficient approach that does not need many learner feedbacks. Like (Iglesias et al., 2009) reinforcement learning is used for filtering presented documents and information to learner, in this system, the WAIR (Web Argent for Information Retrieval) is used.

This architecture includes learner interface agent, information filtering agent and information retrieval agent, and with using search engines and learner profiles, it receives documents for learners. The main point of this system is constructing and updating learner profile. The profile at first is made of some key words, such as learner inputs and general learner characteristic, like language, educations, intelligence and other things which is gotten by him/herself or by his/her friends. These key words are updated during learner and system interaction and by receiving learner feedbacks. Updating includes adding new words to profile, omitting some key words and changing learner profile key words weight. Formally, learner profile is a vector of weight like the following vector:

$$w_p =< w_{p,1}, w_{p,2}, ..., w_{p,n} > \quad (1)$$

$w_{p,k}$ is equal to weight of word "k" and $\|w_p\| = 1$. According to the profile, WAIR sends a query to search engines that every word existence probability is related to its weight in profile after receiving documents. Rank of each document like "I" for profile "p" calculates according to two vector cosine. For system learning, implicit and explicit feedbacks is received. Implicit feedback $R_E(i)$ that is received at the beginning of system work is the points learners give to documents. Explicit feedbacks $R_I(i)$ includes a learner's study time, and the links is followed by learner. Reward for each document is made of the combination all this rewards:

$$r_i = \delta R_E(i) + (1-\delta) R_I(i) \quad (2)$$

Based on this reward, learner's profile updates as follow:

$$w_{p,k}^{(i+1)} = w_{p,k}^{(i)} + \beta r_i I(x_{i,k}) \quad (3)$$

In above formula, $I(X_{i,k})$ is a threshold function that its output is 0, 1 and -1. After results are gained, contents are revised from the point of LLL rules. If any content contravenes LLL rules, they will be omitted. LLL rules made of learning rules and teaching methods is proposed by teacher according to learning map. Some sample of LLL rules come as follows:

> LLL rule validation (learner's profile, content profile) (If (level of content i = A) and (intelligence of learner = 40), then reject content; If (language of content I = "English") and (language of learner = "Farsi"), then reject content; If (mastery level of learner = A) and (Date of publishing content = 1990) then reject content.…)

This conversational approach, which iterates until learners correct answers number, is more than specific threshold. If a learner's correct answers are more than specific threshold, the system sends a message of ending learning process to the learner.

In addition, recommendations are gained from learner search, by investigating friend uses and finding similarity between learner and their friends, so recommendations are produced based on CF (collaborative filtering). An important point in CF used in our approach is the way of weighting to friends recommend. As mentioned before, learner weights are kept in one place, and at time of using CF, these weights are used for assigning similarity. Like previous way, the results of this approach are checked by LLL rules. Another list belongs to teacher recommendations. The teacher, according to his/her content and learners' recognition, recommends to learner. These recommendations are checked by LLL rules to minimize human errors. Relations with the recommendations are checked are as in Table 1.

Table 1

Relations With the Recommendations

Search items = {I_i, I_j, …, I_m} Professor proposal = {I_i, I_j, …} Freinds popular items = {I_a, I_r, I_o, I_j, I_c, …, I_s}	Input
$U_{filtering}$ = SI ; Filtering results = LLL rule validation { Filtering($U_{filtering}$, User profile)} CF results = LLL rule validation {Collaborative filtering (Friends popular items, Friends weight) = CF(FPI, FW)} Professor proposal result = LLL rule validation {Professor proposal}	Process
Rec results = (Filtering results, U CF results, U Professor proposal results} = { I_N, I_{N-1}, I_{N-2}, …, I_0}	Output

Output phase. In this phase, if the value of recommendation (output of similarity recognizer unit) is bellow 2, the teacher should assign that learning content is proper or not. But if the value of recommendation is more than 2, validation is not necessary. The value 2 is an empirical quantity and has been assigned for system efficiency. The snapshot of recommendation results is shown in Figure 6.

Figure 6. Sanpshots of recommendation results.

Feedback processing phase. This phase happens when a learner has studied learning content. System can update itself and improve its recommendations by gathering learners' feedback and analyzing it, updating learner model if it is necessary and design new learning map for learners. Feedbacks include information, such as paper level (from A to Z), edition type (weak 0, excellent 100), recommendation precise (from 0 to 100), usefulness percentage (from 0 to 100) and other things are mentioned by learners. Also, learners can annotate or summarize the content that has been studied. Every annotation is saved with its author name and is useful for other learners that want to study those papers. A learner's feedbacks make it possible to update weight of his/her friends. As friends have an important role in quality of system recommendation and modeling, by comparing learner model and other members, the system recommends most similar members to the learner as a friend, but the learner can accept or reject it.

Conclusion and Future Work

Current LLL systems have been focusing on the interrelations between learners and the system. The LLL systems must track and detect learners' need, and according to this need, adapt itself to make more personalization. Based on LLL systems nature, this paper proposes novel method that provides powerful content recommendations in critical and unknown situations. In addition, three methods are discussed for improving system tolerance in the condition which system has information leakage. In this paper, we try to introduce compatible method for social networks that reuse their information, such as profile, friends and behavior, and so on. In this work, we use open LMS that help its learners to keep up to date to the dynamics of information on the web. Currently, we focus on testing proposed systems and gathering statistical data for evaluation.

References

Bobadilla, J., Serradilla, F., Hernando, A., & MovieLens. (2009). Collaborative filtering adapted to recommender systems of e-learning, knowledge-based systems. *Journal of Knowledge-Based Systems, 10*(1016).

Basu, C., Hirsh, H., & Cohen, W. (2001). Technical paper recommendations: A study in combining multiple information sources. *Journal of Artificial Intelligence, 1*, 231-252.

Bollacker, K. D., Lawrence, S., & Giles, C. L. (1999). A system for automatic personalized tracking of scientific literature on the web. *ACM Conference on Digital Libraries*, 105-113.

Drachesler, H., Hummel, H., & Koper, R. (2008). Personal recommender systems for learners in lifelong learning networks: Requirements, techniques and model. *International Journal of Learning Technology, 3*(4).

Iglesias, A., Martinez, P., Aler, R., & Fernandez, F. (2009). Reinforcement learning of pedagogical policies in adaptive and intelligent educational systems. *Journal of Knowledge-Based Systems, 10*(1016).

Joachims, T., Freitag, D., & Mitchell, T. M. (1997). Web watcher: A tour guide for the world wide web. Proceedings of *International Joint Conference on Artificial Intelligence*.

Kardan, A. A., & Einavypour, Y. (2008). *Eliminating anomalies in learner modeling using two-partial learner model*. ICEIT'08, IAENG.

Kardan, A. A., Speily, O. R. B., & Modaberi S. (2009). Recommender systems for smart lifelong learning systems. *ICVL09 International Conferences on Virtual Learning*, Bocharest, Romani, 2009.

McNee, S. M., Albert, I., Cosley, D., Gopalkrishnan, P., Lam, S. K., Rashid, A. M., …, & Riedl, J. (2002). On the recommending of citations for research papers. In Proceedings of *ACM International Conference on Computer Supported Collaborative Work*, 116-125.

Santos, O. C., & Boticario, J. G. (2008). Recommender systems for lifelong learning inclusive scenarios. *ECAI 2008-Workshop on Recommender Systems*, Patras, Greece.

TANG, T., & Mccalla, G. (2007). *Smart recommendation for an evolving e-learning system*. Canada: Dept. of Computer Science, University of Saskatchewan.

Taghipour, N. (2004). Presenting a hybrid web recommendation system based on web mining. (Master Thesis, Amirkabir University of Technology)

Woodruff, A., Gossweiler, R., Pitkow, J., Chi, E., & Card, S. K. (2000). Enhancing a digital book with a reading recommender. In Proceedings of *ACM CHI*, 153-160.

ZHANG, B., & Seo, Y. (2001). Personalized web-document filtering using reinforcement learning. *Applied Artificial Intelligence, 15*(7), 665-685.