A Multi-Dimensional Recommendation Framework for Learning Material by Naive Bayes Classifier

Mojtaba Salehi

Department of Industrial Engineering
Faculty of Engineering, Tarbiat Modares University
Tehran, Iran
mojtaba.salehi@modares.ac.ir

Isa Nakhai Kamalabadi

Department of Industrial Engineering
Faculty of Engineering, Tarbiat Modares University
Tehran, Iran
nakai@modares.ac.ir

Abstract—**Personal Learning Environment (PLE) solutions can empower learners to design ICT environments for their activities in different learning contexts. Recommender systems have been used for supporting learners in PLE-based activities. Since, in the current recommendation approaches, multidimensional attributes of resource and dynamic interests and multi-preference of learners are not fully considered simultaneously, this paper proposes a novel resource recommendation framework in order to personalize learning environments. Learner Tree (LT) is introduced to take into account the multidimensional attributes of resources and learners' rating matrix simultaneously. In addition, a forgetting function also is used to reflect dynamic preference of a learner and a Bayesian classifier is used to predict rate of unrated resources. The main contribution of this paper is proposing a multidimensional data model to consider multi-preference of learner and using naive Bayes classifier to improve the quality of recommendation in the terms of precision, recall and also intra-list similarity. In addition, the proposed approach tries to satisfy the learner's real learning preference accurately according to the real-time up dated contextual information.**

Keywords- Adaptive recommendation, Multidimensional recommendation, Learning resource, e-Learning, Collaborative filtering

I. INTRODUCTION

In PLEs, personalized recommendations help filtering information based on "soft" but significant context boundaries [1], giving learners the opportunity to take the best of an environment where shared content differed in quality, target audience, subject matter, and is constantly expanded, annotated, and repurposed [2].

In the recent years, several researches have addressed the need for personalization in the web-based learning environment. Addressing more learner-centric technology-enhanced learning (TEL) streams,

recommendations seem to be a powerful tool for personal learning environment (PLE) solutions [3]. The task of delivering personalized learning resource is often framed in terms of a recommendation task in which a system recommends items to an active user [4]. Several educational recommender systems have been proposed in the literature that the most of them focus on recommending suitable resources or learning activities [5]. On the other hand, in PLEs, personalized recommendations help filtering information based on "soft" but significant context boundaries [1]. Locating the suitable learning resources has become a big challenge. One way to address this challenge is the use of recommender systems [6].

Recommender system is being deployed in more and more e-commerce entities to best articulate and accommodate customers' interests. Recommender systems in learning environment need to satisfy other requirements than recommenders in e-commerce. It is necessary that the recommended items be fit into the current context of the user, considering e.g. her competences and learning goals to support her while learning. Approaches that deal with those requirements are comprehensively described by Manouselis et al. [23].

recommendations in time when learners' current interests are changing. Thus, it will lead to a great difference between recommended resources and learners' actual preferences. Therefore, it is necessary to implement an adaptive approach for producing recommendations.

To model adaptive multi-preference of learner, this

Table 1. Overview of the recommendation strategies
(Input, Output, User modeling, recommendation process and some of researches)

Strategies	Input	Output	User modeling and recommendation process	Some of researches
Data mining	Access history of learners	materials, related links, learning activities,	Learner is modeled as sequences of visited resources by gathering information about the learner behavior and recommendation is made using sequential pattern mining and discovering suitable rules for recommendation.	[7-10]
Content-based filtering	Access history and learners rating,	materials, related links, learning activities,	It extracts features of items and generates a classifier that can fit learner's rating behavior. User model includes preference information of learner about item's features.	[9, 11]
Collaborative filtering	Attributes of material and learner	materials, related links, learning activities,	Usually each learner is modeled by a rating vector. Learners with similar rating are identified as neighbor and learners' behavior is extrapolated from their rating.	[12- 17]
Hybrid approach	Access history and learners rating, attributes of material and learner	materials, related links, learning activities,	This approach usually combines rating of learner, access history of learner and attributes of resource and learner for learner's rating prediction or recommendation. User modeling is different in related works.	[18-22, 9]

The objective of TEL is design, develop and test socio-technical innovations for enhancing learning practices and knowledge sharing of individuals and organizations. In addition, by the increasing use of Personal Learning Environments, the TEL field has become a promising application area for recommender systems to suggest most suitable learning resource. The motivation for any recommender system (RS) is to assure an efficient use of available resource. By using this approach we can improve a personal learning path according to pedagogical issues and available resource. As the amount of learning resources is massive meanwhile the learners' preference and relevant contextual information will change dynamically; there will be several drawbacks when applying existing recommendation algorithms to learning environments. 1) Most of existing resource recommendation approach only use learners' rating information and therefore only certain resources which have similar rating by neighborhood users could be recommended. Therefore, to include other suitable information and improve recommendation, we will develop a framework that considers multi-preference of learner based on multi-attribute of resources. 2) The learners' preferences will be changing dynamically. However, as most of existing recommendation algorithms couldn't take into account the access time of historical records, they cannot make good

paper proposes a new resource recommendation framework and relevant recommendation algorithm for learning environments. Therefore, the originality of this paper is that it can take into account the dynamic interests and multi-preference of learners and multidimensional attributes of resource simultaneously. This research can calculate relevant degree between learners and resources based on the target learners' historical access records and multi-dimensional attributes of learning resources. In order to reflect learners' complete spectrum of interests, a multidimensional data model as a Learner Tree (LT) is introduced to consider resources' multi-dimensional attributes information, learners' rating information for modeling of multi-preference of learner. Truly, LT is build based on target learners' historical access records and multi-dimensional attributes of learning resources. Then, a new measure is introduced that can use LT and multi-dimensional attributes of a learning resource for calculating relevance degree between them. In addition, a Bayesian classifier is used to predict rate of unrated resources and a forgetting function also is used to reflect dynamic preference of a learner. The rest of this paper is organized as follows. In Literature review section, the previous related works on learning resource recommender systems is discussed. Methodology section introduces the overall system framework and describes the proposed algorithm step by step. Implication section applies the proposed algorithm for a dataset to evaluate and analyze the performance and limitations of it. Finally,

Conclusion section provides the concluding remarks along with suggestions for future works.

II. LITERATURE REVIEW

Personal learning environments (PLEs) refer to a set of learning tools, services, and artifacts gathered from various contexts to be used by the learners Henri et al. [24]. PLE solutions should provide facilities for empowering learners in using this kind of technology. One possible solution is the application of recommender technology [25].

Recommendation in a TEL context has many particularities that are based on the richness of the pedagogical theories and models [26]. For instance, for learners with no prior knowledge in a specific domain, relevant pedagogical rules could be applied [27]. Different from commercial transaction, learners achieve different levels of competences that have various levels in different domains. Therefore, identifying the relevant learning goals and supporting learners in achieving them is very important. On the other hand, depending on the context, some particular user task may be prioritised. This could call for recommendations whose time span is longer than the one of product recommendations [28]. In addition, in the teacher-centered learning context, one of the most important tasks that needs to be supported is lesson preparation. Lesson preparation can include a variety of information seeking tasks, such as finding content to motivate the learners, to recall existing knowledge, to illustrate, visualize and represent new concepts and information [29]. User modeling and recommendation process of the recommendation strategies and some of related works were presented in the Table 1.

1) Data mining

These techniques usually model learners using the gathered information about the learner behavior, such as navigation history, to produce recommendations. These techniques usually are used for recommendation an effective learning path. Romero et al. [10] developed a specific web mining tool for discovering suitable rules in recommender engine. Their objective was recommendation to a student the most appropriate links/webpages to visit next. In these two researches, learners are modeled as sequences of visited resources and usually association rules are implemented for discovering suitable rules for recommendation. This strategy cannot model multidimensional attributes of learning resources for improving the accuracy of recommendation. Also this strategy usually doesn't use rating matrix as suitable information for recommendation.

2) Content based filtering

This strategy extracts features of items and builds a matching model for them. User model includes preference information of learner about item's features. Since this strategy cannot model similarity between learners for improving the quality of recommendation, only certain resources which are similar to learner's historical preference can be recommended.

3) Collaborative filtering

Collaborative filtering systems that are also called clique-based systems assume that users who had similar choices before will make same selection in the future. Initial hints of relating collaborative filtering to education have appeared in early relevant papers [30]. Soonthornphisaj et al. [16] used Collaborative filtering for resource recommendation. First, the weight between all users and the target learner is calculated by Pearson correlation. Then, the n users with the highest similarity to the active learner are selected as the neighborhoods. Finally, using the weight combination obtained from the neighborhood, the rating prediction is calculated.

4) Hybrid approach

Most of researchers used hybrid approach for resource recommendation to address drawbacks of previous strategies. A proposed system that adopts a hybrid approach for recommending learning resources is the one recently proposed by Drachsler et al. [31].
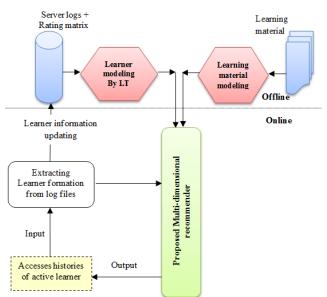

Figure1. Overall architecture of the proposed recommender system to personalize learning environment

The authors proposed a system that combines social-based with information-based recommendation techniques.

In addition, in the hybrid model based recommendation strategy, some of other techniques such as Case-Based Reasoning [32], intelligent agents [33], neural network [34], genetic algorithm [35] and ontology [36] are used. Usually these recommendations require intensive computation. An appropriate recommendation technique can be chosen according to pedagogical reasons. These pedagogical reasons are derived from specific demands of lifelong learning [37]. Therefore, some recommendation techniques are more suitable for specific demands of lifelong learning than others. One way to implement pedagogical decisions into a recommender system is to use a variety of recommendation techniques in a recommendation strategy. In this paper, in order to improve the learning resource recommendation efficiency, a framework for integrating contextual information including multi-dimensional attributes of resources, learner's rating information and time-variant multi-preferences of the learner has developed. Most of researches only use some of this information in resource recommendation process, but our proposed framework can use this information simultaneously to model adaptive multi-preference of learner and improve quality of recommendation.

III. METHODLOGY

As Figure 1 shows in the offline mode, the learner model and resource model are built. Resources are modeled according to content modeling approach that will be explained in the next section. For learner modeling, server usage logs of learners are collected in a certain period. Then, Learner Tree (LT) is built as a multidimensional data model for each learner. In addition in this phase, Naive Bayes (NB) classifier is used for prediction the rates of unrated learning resources and Graduation Forgetting Function (GFF) is implemented to model learners' dynamic preferences. In the online phase, system can calculate relevance degree between the active learner and resources and make suitable recommendation.

A. Offline phase

Learner and learning material modeling are done in this phase.

1) Learning material modeling

We can consider some attributes for learning resource such as subject for example literature, mathematics and computer science. In addition, since ratings of learner's accessed resources that have certain attributes indicate the importance of these attributes for the learner; it can be considered as base for weighting of attributes for the learner. Therefore, in order to consider learner's preference accurately, the multidimensional attributes of learning resource should be considered in the calculation of relevant resources for the learner. It must be noted, this research introduces a multidimensional attribute-based framework for recommendation that involves attributes of resource in the recommendation process, but selection of appropriate attributes may vary in the different systems. System developer can use Learning Object Metadata (LOM) to select suitable attributes. Relevant attributes of learning resource including type of object; author; owner; terms of distribution; format; and pedagogical attributes, such as teaching or interaction style could be selected. In this research according to the simplicity and usefulness and also expert opinions in the education area, we select four attributes including: subject, secondary subject, education type (Bachelor Level (B.L.), Master Level (M.L.), PhD Level (PhD.L.)) and publisher of resource. The most important of attributes (properties) of a learning resource for interest modeling of learner are subject and secondary subject; therefore in this research we use a weighting approach that allocates more weight for subject and secondary subject and less weight for other attributes.

The resource attributes' description model can be defined as a multidimensional attributes vector $M =< (AK_1, AW_1), (AK_2, AW_2),...,(AK_m, AW_m) >$ where AK_t denotes the t-th dimensional attribute's name of resource, AW_t denotes the relevant weight value $AW_1 \geq AW_2 \geq ... \geq AW_m$ and $\sum_{i=1}^{m} AW_i = 1$. Based on this
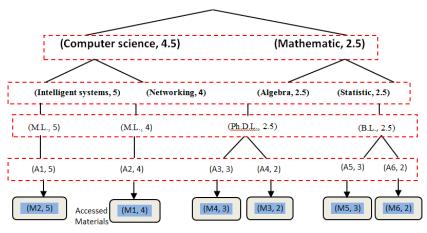

Figure 2. Learner Tree sample

description model, the attributes of a certain resource M_j can be defined as $MA_j = [AK_{j1}, AK_{j2}, ..., AK_{jm}]$ where AK_{jt} denotes the t-th dimension attribute's keyword of M_j. For example:

$$M_i = [(Mathematic, 0.4), (Statstic, 0.25), (Master\deg ree, 0.2), (Author5, 0.15)]$$

The weight of each attribute is produced by system developer to indicate the importance of each attribute. It must be noted new resources can added to the repository in offline mode and a system developer must determine MA_j for each resource, but the relevant weight values are similar to the other resource.

2) Learner modeling

For learner modeling, two types of information are used. (1) Rating is used for modeling of learner's preferences (2) access order of material is used for modeling dynamic interests of learner. In the following of this section, at first, we explain how to use the information in the learner modeling, and then the learner's model is built based on the two types of information.

In this research, learner L_i is defined as a tree with a (m+1)-level in which m indicates the number of attribute of resources. In this tree, the leaf node which represents an accessed resource of L_i is defined as a two-tuple: LT_{leaf} = {*MID, MR*}, where *MID* indicates accessed resource ID, *MR* indicates rating of L_i to a certain resource. The non-leaf node can be defined as a two-tuple: $LT_{nonleaf}$ = {*KA, MR*}, where *KA* is the keyword of the *level*-th attribute of resource. A four-dimension attributes description model based LT is considered in this research including: subject, secondary subject, education type and publisher that a sample is shown in Figure 2.

The *MR* of non-leaf node *k* can be calculated as the mean of *MR* value of all leaf nodes which belong to *k*'s sub-tree. Each accessed resource corresponds to a unique path from root to relevant leaf node, and the keywords of all nodes located in this path correspond to the relevant keywords of M_j's attributes.

A new user must first register and get an ID, then system can build a *LT* for him/ her. It must be noted building and updating trees for different systems may be different, for a huge number of users, system developer must construct a system that updates *LTs* in offline, otherwise for getting better recommendation; system can update *LTs* in online. System can update *LT* based on the following strategy:

Search the keywords of the latest accessed resource attributes ($MA_j = [AK_{j1}, AK_{j2}, ..., AK_{jm}]$) in *LT* from the upper row to the bottom row. If the keyword of *i*-th attribute cannot be matched, the $m-i+1$ new level with latter $m-i+1$ attributes of resource will be created at a new rout and updated *MR* in the whole of tree.

3) Rating prediction

Dependency of our approach on learner ratings can be a drawback. Because several learners must evaluate each learning resource and new resources cannot be recommended until some learners have taken the time to evaluate them. This problem referred to as 'data sparsity' and 'cold start problem. For solving this problem and improving the accuracy of recommendation, we used a predictor for rating prediction unrated resources. By a predictor such as the naive Bayes classifier, we can fill in the missing values of the rating matrix to form a pseudo rating matrix.

Naive Bayes algorithms have frequently adopted the in recent works for information retrieval and recommender systems [38]. The algorithm's popularity and performance for information retrieval and recommender systems applications have prompted researchers to empirically evaluate and compare different variations of naive Bayes that have appeared in the literature [25, 39]. In addition, the naive Bayes classifier is robust with respect to missing feature values, which may make it appropriate to the rating prediction. In this classifier, assuming the features are independent given the class, the probability of a certain class given all of the features can be computed, and then the class with the highest probability will be classified as the predicted class.

In this research, we independently learn one classifier for each learning resource *y*. We train the classifier for learning resource *y* using all learners *u* in the data set who has supplied a rating for learning resource *y*. The input vectors used to construct the classifier for resource *y* consist of ratings for all resources other than resource *y*. We will refer to resource *y* as the class resource, and the remaining resources as feature resources. We can express this naive Bayes classifier for resource *y* in terms of a Bayesian network as seen in Figure 3 as the nodes represent random variables corresponding to the class label M_y, and the components of the input vector $M_1, M_2, ..., M_{y-1}, M_{y+1}, ..., M_m$.

To learn the naive Bayes rating predictor we must estimate $P(M_y = v)$, the prior probability that resource *y* takes value *v* in rating and $P(M_j = w | M_y = v)$, the
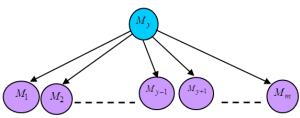

Figure 3. Naive Bayes classifier for rating prediction.

probability that resource *j* takes value *w* in rating given the value of resource *y* is *v*. These probabilities can be estimated using frequencies computed from the training data as seen in Equations 1 and 2.

$$P(M_y = v) = \frac{\sum_{l=1}^{N} \varphi(m_y^l, v)}{N} \qquad (1)$$

$$P(M_j = w \mid M_y = v) = \frac{\sum_{l=1}^{N} \varphi(m_j^l, w)\varphi(m_y^l, v)}{\sum_{w=1}^{V} \sum_{l=1}^{N} \varphi(m_j^l, w)\varphi(m_y^l, v)} \quad (2)$$

Where

$$\varphi(m_y^l, v) = \begin{cases} 1 & if \quad m_y^l = v \\ 0 & otherwise \end{cases}$$

$$\varphi(m_j^l, w) = \begin{cases} 1 & if \quad m_j^l = w \\ 0 & otherwise \end{cases}$$

Where m_y^l is the rate value of resource y by learner l. It must be noted that N is number of learners and V is number of classes in rating (i.e., possible values for rating, 1–5 in this research).

To predict the value of m_y^a given the profile m^a of a particular active learner, we apply a prediction rule to allow for missing values. This prediction rule is shown in Equation 3.

$$\hat{m}_y^a = \arg \max_v P(M_y = v) \prod_{w=1}^{V} P(M_j = w \mid M_y = v) \quad (3)$$

By using above approach for rating prediction, we can make a pseudo rating matrix and improve the quality of recommendation.

B. Online phase

The development of recommendation is done in this phase. It includes gathering the access history of learners and calculating relevance degree between learner and resources and finally selecting the top-N recommended resources.

1) Relevance degree calculation

The preference of a learner may change and the history records couldn't entirely reflect the whole preference of a learner. To have a personal learning environment, inspired from Chuan-chang [40], Gradual Forgetting Function (GFF) concept is introduced in order to reflect dynamic interests and preference of learner's more accurately.

The main idea of GFF is to introduce a forgetting function into calculation of the relevance degree between learner and resources. If the relevance degree between user L_i and M_j is $RD(L_i, M_j)$, then the new relevance degree will be $h(M_j) \times RD(L_i, M_j)$. In this research, we consider a nonlinear forgetting function as follows:

$$h(M_j) = \exp \lambda(x(M_j) - 1) \quad x \geq 1, \quad (4)$$

Where $x(M_j)$ is the access order for M_j by learner L_i. Therefore, the effect of M_j to L_i 's future interest will becoming smaller with resource access process going on, therefore, $h(M_j)$ should be attenuated gradually. In $h(M_j)$, λ is an adjustable parameter used to describe the change rate of user's preference, and the bigger of it, the quicker of the forgetting.

To calculate relevance degree between learning resources and a learner, at first according to each dimensional attribute's keyword, certain resource M_j

is matched with each level's k value of LT from top to down. The matching path of M_j and $LT(L_i)$ denoted as $MP(L_i, M_j)$ is the longest path, where keywords of all nodes of this path are identical to corresponding keywords of resource's attributes. The relevance degree between resource M_j and learner L_i can be calculated by the following equation:

$$RD(L_i, M_j) = h(M_j). \sum_{s \in MP(L_i, M_j)} MW_s.MR_s \quad (5)$$

It must be noted for unrated resource, $h(M_j)$ value is mean of $h(M_j)$ values of rated resources for each learner. MR_s is learner rating information in the s-th level's matching; MW_s is the s-th level's matching weight of LT. since the MW_s should increase with depth growing, in this paper, it can be defined as $MW_s = AW_s^{-1}$.

2) Recommendation

When a learner registers and investigates the resource repository, by using resource accessed history of learner, the learner tree is constructed and evoluted gradually. Then, when a learner requires resource recommendation, system calculates $RD(L_i, M_j)$ based on Equation 5. Finally, the Top-N resources with largest $RD(L_i, M_j)$ are regarded as the Top-N resource recommendation

We know the aim of learning analytics is to determine the specific learning parameters for predicting some specific learning outcomes. In this research also we modeled some features of learning resource and learners as the specific learning parameters to predict interest of learner in a multidimensional space and dynamically.

IV. EXPERIMENTS

In this section, a set of experiments have been conducted to set parameters and examine the effectiveness of our proposed recommender system in terms of recommendation accuracy and quality.

A. Experiment environment

In the existing state-of-the-art, recommender systems use one of the following approaches to evaluate the performance of their systems: a real environment, an evaluation environment, the logs of the system or a user simulator.

By an evaluation environment, we let a set of users to interact with the system over a period of time. In this approach, usually, the results are not reliable enough because the users often know the purpose of the evaluation. The analysis of the logs files of real users obtained in a real or evaluation environment is also a common technique for evaluation of recommender systems. In this condition, the cross-validation technique is often used for evaluation of recommender system. A few systems use simulated users to evaluate their performance. This approach can enable large-scale experiments to be implemented quickly and also makes that experiments are repeatable and perfectly controlled. However, the

main drawback of this system is that it can't simulate the real behavior of a user. Users are too complicated to predict and also their feelings and their emotions, and therefore, their actions change dynamically. Results obtained in a real environment with real users are the best way to evaluate a recommender system. But the main problem of the real and the evaluation environments is that repetition of the experiments. Therefore, in this research we use log files of real environment that enables us to repeat the experiments and implement the cross-validation technique.

In this research, a real-world dataset, learning records are applied in our experiments. The learning records dataset comes from the usage data of the course management system Moodle. MOODLE (Modular Object-Oriented Dynamic Learning Environment) is defined as a course management system (CMS), a free, Open Source software package designed using pedagogical principles, to help educators by creating effective online learning communities. Moodle stores detailed records of students' activities and the educator can access summarized reports about these activities according to the categories specified by the Moodle system.

The used dataset contains 59581 lending records from 1723 users on 32345 books where each record contains timestamp and rating information (as the ratio of certain lending time segment to maximum lending time segment), in addition it contains books' type information and users' basic information. The characteristics of this dataset are summarized in Table 2. In order to increase the number of records in test set as much as possible so as to eliminate the effect of accidental factor, the top 60% access records of each user in ordered dataset are used as training set and the remnant 40% access records are used as test set.

Table 2. The characteristics of the dataset used in experiments

Number of users	Number of items	Number of transactions	Density	Average person' records
1723	32345	59581	0.107%	34.58

Moodle has a number of interactive learning activity modules such as forums, chats, quizzes and assignment. In addition, by Moodle we can register and track user's accesses (user identification, IP and time) and the activities and resources that have been accessed by user. Therefore each record includes (1) IP address, (2) Date and hour of the access, (3) Complete name of the user who has been accessed, (4) Type of access (resource view, course view, etc.), (5) Specific element of the course that has been visited. According these features, this dataset is suitable for evaluation of proposed approach.

B. performance metrics

Precision and Recall are popular, well-established metrics from the information retrieval community. Since the test data set or relevance set is the real tracking of users in the learning environment, truly by using this approach for recommendation, we want to predict the real behaviors of users in the learning environment. Despite of this approach takes into account the dynamic behavior of users for acquiring better recommendations, it is a prediction problem again, and thus Recall and Precision measures are valid for this type of dynamic data also.

The precision and recall are most popular metrics for evaluating information retrieval system. For the evaluation of recommender system, they have been used by various researchers. When referring to Recommender Systems the recall can be defined as follows:

$$Recall = \frac{|test \cap top - N|}{|test|} \qquad (6)$$

Where $top - N$ denotes the recommendation set and $test$ denotes the test set. The precision when referring to recommender systems can be defined as follows:

$$Precision = \frac{|test \cap top - N|}{N} \qquad (7)$$

Where N denotes number of recommendation.

Since usually there is a trade-off between algorithm running time and recommendation precision, we measure mean running time for single user that can help us for an appropriate decision.

Users interact with recommendation list and accuracy metrics cannot see this problem because they are designed to judge the accuracy of individual item predictions; they do not judge the contents of entire recommendation lists. Since the recommendation list should be judged for its usefulness as a complete entity, not just as a collection of individual items, in this research we also define an Intra-List Similarity Metric [41] as follows:

$$ISM(List) = \frac{\sum\limits_{R_i} \sum\limits_{R_j, i \neq j} f(M_i, M_j)}{\binom{|List|}{2}} \qquad (8)$$

Where

$$f(M_i, M_j) = \frac{mat(M_i, M_j)}{m} \qquad (9)$$

Where *mat* indicates number of matching attributes between resource M_i and M_j. As it was said before m is number of considered attributes for resource. Higher similarity denotes lower diversity.

C. Parameter setting

Firstly, we will analyze how λ affects the recommendation performance in order to determine the values of this parameter (as different dataset may correspond to different optimal value of this parameters, this analysis process is just based on records dataset). Figure 4 shows the impacts of λ on the precision of the proposed approach while N=30, p=300, $K = 30$. It indicates that the best precision can be obtained with $\lambda = 0.3$.

D. Performance evaluation

In this section, the architecture proposed and corresponding recommendation mechanism is compared with traditional recommendation algorithms including vector space model-based content-based recommendation algorithm [42], user and item combined collaborative-based recommendation algorithm [43] and hybrid recommendation algorithm [44]. In relevant input parameters, N denotes the number of recommendation resources and p denotes the number of participated users which are selected from dataset to build experiment dataset. Since the main objective of this research is to recommend the suitable resource for learners, in the following of this section we conduct some of experiment to compare result of recommendation for the proposed method for different situation. Therefore, the precision, recall and mean absolute error of proposed method are compared with the three traditional algorithm based on the number of participated users, p and number of recommendation resources, N.
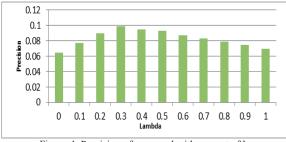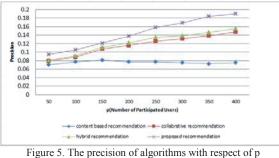

Figure 4. Precision of proposed with respect of λ

The precision of algorithms with respect of *p*: In the first experiment, the precision of recommendation algorithm is compared with respect to p that denotes the number of participated users which are selected from records dataset to build experiment dataset while N=30. As shown in Figure 5, with increasing of p, the precision of algorithms is increasing except content based algorithm, where the proposed recommendation algorithm always produces better performance than any other algorithms. When p is small, as users' information cannot be utilized efficiently, the collaborative based algorithm cannot detect effective similar users and will lead to bad result. With p increasing, by utilizing much more users' information efficiently, the performance of collaborative based algorithm will be enhanced gradually. However, improved hybrid recommendation algorithm as combination of content-based and collaborative-based recommendation algorithm is somewhat better than either content-based or collaborative based algorithm alone. also, the proposed recommendation algorithm that can take into account the time-related dynamic preference, multi-dimensional attributes of resources and users' rating information synthetically, the better results can be obtained whenever the number of p is small or large.

Figure 5. The precision of algorithms with respect of p

The precision of algorithms with respect of *N*: In the second comparison experiment the precision of recommendation algorithms is compared with respect to N (the number of recommendation resources) while p=300. As shown in Figure 6, with the increasing of N, the precision of algorithms is decreasing. However, the proposed algorithm always produces better performance than other algorithms, especially when N is small. Because during the changing process, according to precision formula, the numerator and denominator of precision will increase synchronously, but denominator gets the higher increasing rate. In addition, the proposed algorithm by integrating multi-dimensional attributes of resource, access order of resources and also users' rating can reflect the actual preference and interests of users accurately. Based on them, the proposed algorithm can filter and rank the candidate resources much more efficiently.
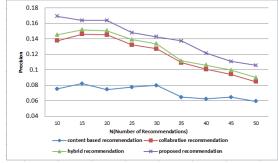

Figure 6. The precision of algorithms with respect of N

The Mean Running Times for single learner of algorithms with respect of *p*: In the third comparison experiment, the mean running times for single learner of all algorithms is compared with respect to p while N =30. As shown in Figure 7 at all times, content-based algorithm is faster than any other algorithms. The running time of the proposed and improved hybrid recommendation algorithms are slightly larger than collaborative based algorithm. According these experiments, although the proposed recommendation will get higher precision in most case, it will cost the largest running time. Therefore, there is a trade-off between algorithm running time and recommendation precision when choosing the proposed recommendation algorithm.

The Recall of algorithms with respect of *p*: In the fourth comparison experiment, the recall of recommendation algorithms is compared with respect to p while N=30. As shown in Figure 8, when p is low, the recall of all algorithms is approximately

equal, but by increasing of *p*, the proposed recommendation has better results. Since learner modeling is based on attributes of accessed learning resources, proposed approach can find more relevant resource for learner and therefore get better result for recall measure. According these experiments, the proposed algorithm recommendation has higher precision and recall in most case.
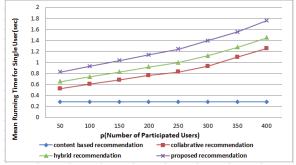

Figure 7. The Mean Running Times for single learner of algorithms with respect of *p*
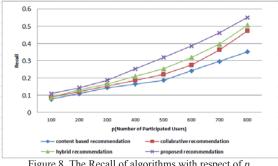

Figure 8. The Recall of algorithms with respect of *p*

The *ISM* of algorithms with respect of *N*: In the sixth comparison experiment, the *ISM* of recommendation algorithms is compared with respect to *N* while *p*=300. As shown in Figure 9, at all time, the proposed algorithm have lower *ISM* than any other algorithms that means higher diversity. Because our proposed approach uses attributes of learning resource in recommendation process. By increasing number of recommendations, diversity decreases for all algorithms. As Figure 9 indicates content-based filtering has the lowest diversity and diversity in collaborative and hybrid recommendation is approximately equal.
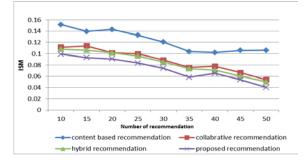

Figure 9. The ISM of algorithms with respect of *N*

V. CONCLUSION

Over the recent years, recommender systems have been successfully deployed in various areas online retailing and social networking. Due to the success of this kind of technology, research on TEL has started to deal with recommender strategies for learning. However, as the repository of learning resources is very massive and learners' preference changes dynamically, there are several drawbacks when applying the existing recommendation algorithms. To address these drawbacks, in this paper, we propose an effective e-learning recommendation framework by modeling adaptive multi-preferences of learner. In the proposed algorithm, in order to improve recommendation accuracy, Learning Tree was introduced that consider multi-dimensional attributes of resources, and relevant rating information simultaneously. The experiment results show that our algorithm can outperform traditional recommendation algorithms significantly and could be more suitable for personal learning environments. Based on the proposed algorithm, the learner's real learning preference can be satisfied accurately according to the real-time up dated contextual information. The main contribution of this paper is proposing a multidimensional data model and using naive Bayes classifier to improve the quality of recommendation in the terms of precision, recall and also intra-list similarity.

For future researches, we can plan to continue our work on multidimensional collaborative filtering and multidimensional hybrid recommender system to further improve the recommendation performance. However, in personal learning environments, there is a temporal-dependency relationship in the learning processes (resource access processes), that can reflect learner's latent resource access pattern and preference. Thus, by mining the learners' historical access records, it can be recommended the most probable resource which will be accessed in near future by the learner for improving the performance of resource recommendation and solving new user problem.

REFERENCES

[1] S. Wilson, P.O. Liber, M. Johnson, P. Beauvoir, and P. Sharples. "Personal Learning Environments: Challenging the dominant design of educational systems," Journal of e-Learning and Knowledge Society, vol. 3, no. 2, pp. 27-28, 2007.

[2] S. Downes, "New Technology Supporting Informal Learning. Journal of Emerging Technologies in Web Intelligence," vol. 2, no. 1, pp. 27-33, 2010.

[3] F. Mödritscher, "Towards a recommender strategy for personal learning environments," Procedia Computer Science, vol. 1, no. 2, pp. 2775-2782, 2010.

[4] B. Mobasher, Data Mining for Personalization. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web: Methods and Strategies of Web Personalization, 2007, pp. 1–46.

[5]   N. Manouselis, H. Drachsler, R. Vuorikari, H. Hummel, and R. Koper, Recommender systems in technology enhanced learning, Recommender Systems Handbook, 2011, pp. 387–415.

[6]   K.I. Ghauth and N.A. Abdullah, "Learning resources recommendation using good learners' ratings and content-based filtering," Educational Technology Research and Development, vol. 58, no. 6, pp. 711–727, 2010.

[7]   E. García, C. Romero, S. Ventura, and C.de. Castro, "An architecture for making recommendations to courseware authors using association rule mining and collaborative filtering," User Model User-Adaption Interaction, vol. 19, pp. 99–132, 2009.

[8]   E. García, C. Romero, S. Ventura, and C.de. Castroa, "A collaborative educational association rule mining tool" The Internet and Higher Education, vol. 14, no. 2, pp. 77–88, 2011.

[9]   M.K. Khribi, M. Jemni, and O. Nasraoui, "Automatic recommendations for e-learning personalization based on web usage mining techniques and information retrieval," Educational Technology & Society, vol. 12, no. 4, pp. 30–42, 2009.

[10]   C. Romero, S. Ventura, A. Zafra, and P. de Bra, "Applying Web usage mining for personalizing hyperlinks in Web-based adaptive educational systems," Computers & Education, vol. 53, pp. 828–840, 2009.

[11]   G. Liang, K. Weining, and L. Junzhou, Courseware recommendation in e-learning system. In Advances in web based learning, 2006, pp. 10–24.

[12]   T. Kerkiri, A. Manitsaris and A. Mavridou, "Reputation Metadata for Recommending Personalized learning resources," Second International Workshop on Semantic Media Adaptation and Personalization, 2007, pp. 110–115.

[13]   D. Lemire, H. Boley, S. McGrath and M. Ball, "Collaborative Filtering and Inference Rules for Context-Aware Learning Object Recommendation," International Journal of Interactive Technology and Smart Education, vol. 2, no. 3, pp. 1–11, 2005

[14]   J. Bobadilla, F. Serradilla, and A. Hernando, "Collaborative filtering adapted to recommender systems of e-learning." Artificial Intelligence (AI) in Blended Learning, vol. 22, no 4, pp. 261–265, 2009.

[15]   O.C. Santos, and J.G. Boticario, Recommendation strategies for promoting eLearning performance factors for all. In The 6th workshop on intelligent techniques for web personalization & recommender systems, 2008, pp. 89–98.

[16]   N. Soonthornphisaj, E. Rojsattarat, and S. Yim-ngam, Smart E-learning using recommender system. Proceedings of the 2006 international conference on Intelligent computing, 2006, pp. 518–523.

[17]   D. Aijuan, and W. Baoying, Domain-based recommendation and retrieval of relevant resources in e-learning. In IEEE international workshop on semantic computing and applications, 2008, pp. 103–108.

[18]   H. Tan, J. Guo and Y. Li, E-Learning Recommendation System. International Conference on Computer Science and Software Engineering, vol. 5, pp. 430–433, 2008.

[19]   A. Klašnja-Milićević, B. Vesin, M. Ivanović, and Z. Budimac, "E-Learning personalization based on hybrid recommendation strategy and learning style identification," Computers and Education, vol. 56, no.3, pp. 885–899, 2011.

[20]   M.A. Otair, and A.Q.A. Hamad, Expert personalized e-learning recommender system. In Proceedings of the First International Conference on E-Business and E-Learning. 2005.

[21]   Y. Zheng, and L. Li, "A three-dimensional context-awareness model for peer recommendation in the e-learning context. International Journal on E-Learning," vol. 7, no. 1, pp. 153–168, , 2008.

[22]   M. Salehi and I. Nakhai Kmalabadi, "A Hybrid Attribute–based Recommender System for E–learning Material Recommendation," IERI Procedia, vol. 2, pp. 565–570, 2012.

[23]   N. Manouselis, R. Vuorikari, and F. van Assche, "Collaborative recommendation of learning resource s: an

[24]   F. Henri, B. Charlier, and F. Limpens. Understanding PLE as an Essential Component of the Learning Process. Proceedings of ED-Media 2008 Conference, Vienna, Austria, 2008, pp. 3766-3770.

[25]   P. Resnick, and H.R. Varian. Recommender systems. Communications of the ACM 40, 1997, pp. 56-58.

[26]   D. Lewis, Naive (Bayes) at Forty: The Independence Assumption in Information Retrieval. In: European Conference on Machine Learning, Chemnitz, Germany, 1998, pp. 4-15.

[27]   L. Vygotsky, Mind in Society: The Development of Higher Psychological Processes. Harvard University Press, 1978.

[28]   G. McCalla, "The Ecological Approach to the Design of E-Learning Environments: Purpose based Capture and Use of Information about Learners." Journal of Interactive Media in Education, Special Issue on the Educational Semantic Web, vol. 7, pp. 1-23, 2004.

[29]   F. Ricci, L. Rokach, and B. Shapira, Introduction to recommender systems handbook. Recommender Systems Handbook, 2011, pp. 1–35.

[30]   L. Terveen, W. Hill, B. Amento, D. McDonald and J. Creter, PHOAKS: a system for sharing recommendations. Communications of the ACM 40, 1997, pp. 59–62.

[31]   H. Drachsler, H.G.K. Hummel, B. Van den Berg, J. Eshuis, A. Berlanga, R. Nadolski, W. Waterink, N. Boers, and R. Koper, "Effects of the ISIS Recommender System for navigation support in self-organized learning networks," Journal of Educational Technology & Society, vol. 12, pp. 122–135, 2009.

[32]   A. Gil, S. Rodríguez, F. De La Prieta, J.F. De Paz and B. Martín, "CBR proposal for personalizing educational content, "http://www.scopus.com/source/sourceInfo.url?sourceId=5100152904&origin=recordpage" \o "Go to the information page for this source" Advances in Intelligent and Soft Computing," vol. 152, pp. 115-123, 2012.

[33]   O. Holanda, R. Ferreira, E. Costa, I. I. Bittencourt, J. Melo, M. Peixoto, and W. Tiengo, "Educational resources recommendation system based on agents and semantic web for helping students in a virtual learning environment," International Journal of Web Based Communities, vol. 8, no.3, pp. 333-353, 2012.

[34]   Z. Sevarac, V., Devedzic, and J. Jovanovic, "Adaptive neuro-fuzzy pedagogical recommender," Expert Systems with Applications, vol. 39, no. 10, pp. 9797-9806, 2012.

[35]   M. Salehi and I. Nakhai Kmalabadi, M. B. Ghaznavi Ghoushchi, "A New Recommendation Approach Based on Implicit Attributes of Learning Material," IERI Procedia, vol. 2, pp. 571–576, 2012.

[36]   M. Ferreira-Satler, F.P. Romero, V.H., Menendez-Dominguez, A. Zapata and M.E. Prieto, "Fuzzy ontologies-based user profiles applied to enhance e-learning activities" Soft Computing, vol. 16, no. 7, pp. 1129-1141, 2012.

[37]   H. Drachsler, H.G.K. Hummel, and R. Koper, "Personal recommender systems for learners in lifelong learning: requirements," techniques and model," International Journal of Learning Technology, vol. 3, no. 4, pp. 404-423, 2008.

[38]   X. Su, and M. Khoshgoftaar, "A Survey of Collaborative Filtering Techniques," Hindawi Publishing Corporation Advances in Artificial Intelligence, vol. 4, no. 19, 2009.

[39]   K. Miyahara and M.J. Pazzani, "Improvement of collaborative filtering with the simple Bayesian classifier," Information Processing Society of Japan, vol. 43, no.11, 2002.

[40]   L. Chuan-chang, "Search recommendation model based on user search behavior and gradual forgetting collaborative filtering strategy," The Journal of China Universities of Posts and Telecommunications. Vol. 17, no. 3, pp. 110–117, 2010.

[41]   C.N. Ziegler, S.M. McNee, J.A. Konstan, and G. Lausen, Improving Recommendation Lists through Topic Diversification. In Proc. of 14th international conference on World Wide Web, 2005, pp. 22-32.

[42]   G. Adomavicius, and A. Tuzhilin, "Toward the Next Generation of Recommender Systems: A Survey of the State-

of-the-Art and Possible Extensions," IEEE Transactions on Knowledge and Data Engineering, vol. 17, no. 6, pp. 734-749, 2005.

[43] H. Ma, I. King, and M.R. Lyu, Effective missing data prediction for collaborative filtering. In: Proceedings of the 30th International ACM SIGIR Conference on Information Retrieval. 2007.

[44] P. Baudisch, Joining Collaborative and Content-Based Filtering. In Proceedings of the Conference on Human Factors in Computing Systems. 1999.

Mojtaba Salehi received his B.Sc. degree from Shahid Bahonar Universityin 2004, and the M.Sc. degree from Tehran University in 2006. He is currently a Ph.D. candidate at department of industrial engineering in Tarbiat Modares University. During 2012, he was a researcher in department of mathematics & computer science at the University of Tehran. His research areas of interest include soft computing, data mining and machine learning, recommender systems and applied multivariate analysis.

Isa Nakhai Kamalabadi received his B.Sc. degree from Shahid Beshti University in 1979, the M.Sc. degree from Tarbiat Modares University and the Ph.D. in industrial engineering from Toronto University in 1995. He is currently working as an Associate Professor in Tarbiat Modares University. His research interests include: optimization, operational research, information technology, scheduling & sequencing, supply chain management and logistics.