HYBRID RECOMMENDER SYSTEM FOR E-LEARNING APPLICATIONS

V.Vanitha¹, DR. R. Shanthi²,

¹P.G Student, M.E CSE, Alpha College of Engineering, Chennai, T.N, India.
²Dean, Alpha College of Engineering, Chennai, T.N, India.

vanikkdi@gmail.com, acedean.students@gmail.com

Abstract:

Recommender systems are regularly used in e-commerce applications. These systems combine ideas from machine learning and user profiling to provide users with an information retrieval applications. They can be also used in e-learning and web-based educational systems like LMS. Nowadays, E-learning are widely adopted in educational organisation. The advances in e-learning has changed the traditional way of learning and has provided a positive environment for both learners and teachers. Educators are finding it difficult to select the suitable learning materials for the learners from vast collection of materials available. This paper provides a novel approach to implement a recommender system for e-learning using association rule mining. This system makes recommendations based on learners’ current knowledge.

Keywords—Recommender system, Association rule mining, E-learning.

I INTRODUCTION

Over the past years, Recommender systems are used in various applications to satisfy the needs of the different users in different circumstances. They have been successfully implemented in various domains from e-commerce to search engines.

Different approaches can be used to implement recommender systems. Content-based and case-based recommender systems rely on feature-based descriptions of items as the basis for recommendation. A typical content-based recommender suggests items for user on the ground of past history, which are similar to items that the user has bought in the past.

Collaborative filtering recommender systems are the most commonly used systems. They use the information provided by other similar users to make recommendations to a particular user. This can be compared to when an item is bought based on the suggestion made by a friend. Thus, these systems use the information provided by others in order to tie it to the profile available about specific user to make a prediction.

Collaborative recommender systems commonly use association rules. Though they were originally developed for retail industry, the same idea can be applied to any domain particularly in e-learning. Due to the inherent nature of scalability and interpretability, they provide an excellent basis for a recommender system. The scalability of this algorithm has also been well recorded in literature [1].

The application of recommender in e-learning, particularly the learner-centred approach aimed at improving the efficiency of learning, has series of hurdles that need to be addressed [2]. The foremost hurdle is the that these tools are too complex for educators to use and their features do not cover the scope of what an educator might.

II RELATED WORK

For more than a decade, researches have been carried out on the application of recommender systems in e-learning and also recommender system based on association rule mining.

In [3], the researchers has proposed mining contrast rules that are of interest for web-based educational systems. Contrast rules help them to identify attributes that characterize patterns of performance disparity between different groups of students.

In [4], recommender system for e-learning is implemented based on agent technology. This system which use association rule mining to show associations between user’s action and URL. Based on a learner’s access history, agents recommend online learning materials/activities in a on line course web-site.

In [5], association fuzzy rules is used in a personalized e-learning recommender system. Author implemented the system using fuzzy matching rules. These rules discover associations between a learner’s requirements and a list of learning objects.
In [6], researchers used association rule, symbolic data analysis and SQL queries to mine student data. These data are captured from a web-based tutoring tool. Using these rules, the system finds mistakes that occur together.

Freyberger et al. [7] use association rules to guide a search for the best fitting transfer model of student learning in intelligent tutoring systems. The association rules determine the operation that needs to be performed on the transfer model to predict a student’s possibility of success.

Chang et al. [8] proposed (ew Fast UpdateMethod (NFUP), anApriori-based algorithm. To mine new frequent patterns in updated database, NFUP partitions the database logistically time interval (month or year). For each item, the ending period of exhibition interval is similar. It scans each partition backward and can reveal new frequent itemsets at the latest time periods. It does not even require scanning of the original database every time. The NFUP implements incremental algorithm requiring scanning of newly added items only.

Leung et al. [9] proposed a canonical-order tree named CanTree, which is an extension of the rule mining algorithm FP-tree. It scans database once in the canonical order to acquire the items of the database. Hence, it does not need to search and discover mergeable trails like CATS tree. After the construction of the tree, original FP-growth is applied to mine frequent patterns [10].

III PROPOSED ARCHITECTURE

This section introduces the proposed architecture for developing an efficient recommender system using association rule mining. It aims to design and develop a framework that may efficiently assist the learning management systems. The overall system architecture is shown in Fig. 1.

Databases stores all the information about the user and learning objects. Database layer includes the learners’ current knowledge library, learners’ assessment score, the learning resources’ object database and the rules mined using association rule mining algorithm. The learners’ current knowledge library has every learner’s learning behavior and related properties, including the currently learning knowledge ID, learning level and learning time, etc.; The learners’ assessment score library contains a learner's current state of study, assessment score etc.; The library of learning re-sources’ object is used for the storage of learning objects.

Figure 1: System Architecture

The proposed framework consists of two phases. The first phase is to generate a set of association rules using Apriori algorithm. The second phase is to use the generated association rules on learning object repository to recommend learning objects to a user.

3.1 Phase 1 - Association Rule Generation

First step is to generate association rules using Apriori algorithm. Number of rules generated depends on the minimum support, and minimum confidence. If they are high, there will be few rules. If they are low, there is a danger of many rules which increases the run time. Considering these facts, the minimum support, and minimum confidence are set at 60% and 80% respectively.

3.2 Phase 1 - Evaluation Of Generated Rules

The second part of the first phase implements evaluation of the rule subjectively. It aims to determinethe interestingness of the rules obtained by Apriori algorithm. Interestingness of the rule is measured by the following parameters:

1) Usefulness of the generated rule according to teachers
2) Interestingness of the rule according to experts.

Let \( T_1, T_2, \ldots, T_m \) be \( m \) different teachers, \( R \) the set of expected association rules found by Ti \( (i=1,2,\ldots,m) \), \( R = \{R_1, R_2, \ldots, R_m\} \); and let \( EX_1, EX_2, \ldots, EX_k \), be \( k \) experts. Let \( R = \{R_1, R_2, \ldots, R_n\} \) represent all the rules in \( R \), then the weight of \( R_i \) can be defined as:

\[
W_{teachers R_i} = \frac{\text{NumVotesTeachers}(R_i)}{\sum_{j=1}^{n} \text{NumVotesTeachers}(R_j)}
\]

where \( i=1,2,\ldots,n \) and \( \text{NumVotesTeachers}(R_i) \) is the number of teachers voted for rule \( Ri \).
By applying the same to the experts’,

\[ W_{experts Ri} = \frac{\text{NumVotesExperts}(R_i)}{\sum_{j=1}^{k} \text{NumVotesExperts}(R_j)} \]

where \(i=1,2,...,n\) and \(\text{NumVotesExperts}(R_i)\) is the number of experts voted for rule \(R_i\). Therefore, the weight of rule \(R_i\) can be expressed as a weighted measurement of the votes registered by teachers and experts, so that:

\[ W_{Ri} = W_{teachers Ri} \cdot C_u + W_{experts Ri} \cdot C_e : C_u + C_e = 1 \]

where \(C_u\) and \(C_e\) are the weighted coefficients denoting the opinions of the teachers and experts respectively.

Once the weight of each rule has been measured, an interestingness calculation can be devised, which is weighted accuracy (\(W_{Acc}\)).

\[ W_{Acc R_i} = W_{R_i} \cdot \frac{\sum_{j=1}^{m} accR_{i,j}}{m} \]

where \(W_{R_i}\) is the weight of the rule.

3.3 phase 2 – Recommendation based on rules

The second part is to apply the subjectively selected rules to recommend learning objects. Specifically, the proposed algorithm addresses the recommendation of appropriate learning objects by applying the generated association rules to offer recommendations for the user:

for each target user \(m\) do

find the learning objects into two classes based on the score of user \(m\):

Suitable Items Class (Score \(\geq 65\))
Non-Suitable Items Class (Score \(<65\))

for each item \(n\) in the Suitable Items Class do

if the item \(n\) is in the associated items then calculate interestingness of the item.
For each interested item then recommend associated item \(u\) to the user \(m\).
end if
end for
end for

IV EXPERIMENTAL RESULTS

In this section, we evaluate the performance of the proposed framework. The experiment aims to evaluate

1) Does the recommended learning object satisfy the need of the learner?
2) Whether the contents of the recommended object are helpful for them?

4.1 Experimental setup

Two criteria selected for measurement are: Recall and Precision rate.

For the proposed model, \(Relevant\) is defined as the set of relevant learning objects for a given query and \(Retrieved\) is the learning objects retrieved by the system for the given query.

Recall and precision is defined as follows:

\[ \text{Recall} = \frac{\text{number of retrieved relevant LOs}}{\text{number of all relevant LOs}} \]
\[ \text{Precision} = \frac{\text{number of retrieved relevant LOs}}{\text{number of all retrieved LOs}} \]

Recommended results are evaluated using Mean Absolute Error (MAE). A feedback regarding the recommended learning object is collected from the user. The MAE is defined as the difference between the given feedback and the prediction. It is given as follows:

\[ \text{MAE} = \frac{1}{N} \sum_{i=1}^{N} |p_i - q_i| \]

where, \(N\) represents the number of comparisons. Lower the MAE value, better the prediction accuracy.

4.2 Experimental Results

This experiment observes the variation of preference feedbacks for the learners with different weights, i.e \(w=0.3\) and \(1-w=0.7\), \(w=0.5\) and \(1-w=0.5\), and \(w=0.7\) and \(1-w=0.3\). The results are shown in Figure 1(a & b).

Figure 1(a) displays how soon the learners’ preference patterns are formed. The x axis represents the number of learning objects studied. For example, \((Y=2, X=5)\) means that the average score of preference feedbacks for the first 5 uses of learning objects is 2. The preference MAE has to be observed to understand the reasons that result in poor results of low preference feedback score.
Figure 1. Direct preference feedbacks

Figure 1(b) shows how the preference MAE varies against the number of learning objects studied. All three weight proportions have their average MAEs lower than 1. This means in fact the system can infer learners’ preferences accurately to recommend suitable learning objects. The reason behind the poor performance is probably that the feature values of candidate learning objects are not completely in accord with learners’ preferences.

IV CONCLUSION AND FUTURE WORK

In the near future, E-learning will become more and more popular. Issues have to be solved before a learner can really benefit from the vast amount of E-learning object repositories on the Internet. This recommender system uses association rule mining and collaborative filtering in order to help the teacher maintain and continuously improve e-learning courses. It uses a weight-based evaluation measurement to rank the rules discovered, taking into account the opinion of both experts and teachers to produce more effective recommendations.

For the future work, the possibility of integrating educational preferences in the learner’s model such as learning styles, media types, etc are analysed. The learner’s model should be composed of three components: learner’s profile, learner’s knowledge and learner’s educational preferences. All these components should be detected automatically within e-learning systems. We expect this enrichment of the learner’s model to increase the quality of learning object recommendations especially from an instructional point of view.

REFERENCES


