

Recommendation System for Student E-Learning Courses

K.Nasaramma, M.Bangaru .Lakshmi, G.Prasanna Priya, G.HimaBindu

Asst. Prof.,

Asst. Prof.,

Asst. Prof.,

Asst. Prof.,

Vignan's Institute Of Information Technology, Visakhapatnam, India.

Abstract

Nowadays to develop an e-learning system which provides the learning environment suitable for each individual learner's learning style. In this system, we developed a method to recommend courses that are suitable for a student. A student's course adaptability for a particular course can be estimated based on the result gained from a previous choices which is conducted prior to the beginning of the course by using the multiple regression models that has been derived from the past students' data. The validity of the developed method was confirmed to some extent by applying the method to assess the student's course adaptability for each course. Furthermore, this system explains the functions of the e-learning course recommendation system that can be added by using the method. The traditional system of selecting courses to carry out research work is time consuming, risky and a tedious task, that not only badly affect the performance but the learning experience of a researcher as well.

This approach may be helpful to learners to increase their performance and improve their satisfaction level as well. The proposed recommender systems would perform better by mitigating the weakness of basic individual recommender systems. There are many courses available for students, and sometimes it is hard for a student to perceive information related to those courses and decide which course to take. This work aims to build a system to suggest online courses to users based on their profile and the similarity with other users. For this work three techniques were used to extract the information and suggest online courses: Apriori algorithm By combining these three techniques the system can offer more accurate recommendations and only considers the interests of each user. Thus, users will not feel tired while perceiving information of their interest and will keep engaged and interested to use the system

Keywords: Data mining, Apriori algorithm

EXISTING SYSTEM

Nowadays there are many courses available for students, and sometimes it is hard for a student to perceive information related to those courses and decide which course to take. This work aims to build a system to suggest online courses to users based on their profile and the similarity with other users. For this work three techniques were used to extract the information and suggest online courses: Content Based, Collaborative filtering and Hybrid. By combining these three techniques the system can offer more accurate recommendations and only considers the interests of each

user. Thus, users will not feel tired while perceiving information of their interest and will keep engaged and interested to use the system.

PROBLEMS/ SHORTCOMINGS OF THE CURRENT SYSTEM

The current system presents a number of shortcomings that makes it ineffective.

This include:

- (i) Storage of the records is problematic especially due to volume of data
- (ii) Report generation is very difficult and inaccurate since some files get misplaced.
- (iii) Data is not converted easily to information
- (iv) Unpredictable Items
- (v) Changing User Preferences
- (vi) Readiness of the information.
- (vii) Data handling is a problem
- (viii) Lack of Data: Perhaps the biggest issue facing recommender systems is that they need a lot of data to effectively make recommendations.

PROPOSED SYSTEM

Recently, many aspects of receiving a college education have been changed. The volume of course-related information available to students is rapidly increasing. This abundance of information has created the need to help students find, organize, and use resources that match their individual goals, interests, and current knowledge. One of the concerns students have is to make decisions about which courses to take.

The concern is more serious for graduate students who have more freedom to choose courses while they care more about taking courses that contribute to their progress towards career goals. To make these decisions, they use information from course catalogs and schedules, consult with their advisors and seek guidance from their classmates, especially those with similar interests. To give better decision making support to students who wish to make relevant course choices, we have developed a course recommendation system that recommends courses to students based on other similar students.

Our collaborative recommendation system tries to recommend

e-learning courses to students based on what other similar students have taken. It recommends courses and specifies expected grades for these courses. Accordingly, the student may take a course that is recommended by the system with an acceptable grade, the students take the advices from other students that took such courses. In our recommendation system, we automatically find similar students and then apply association rule mining algorithm on their courses to create courses association rules. Discovered courses association rules are used to get recommendation.

To obtain courses association rules, courses dataset is built by mapping each course either compulsory or elective to an item and each student to a transaction. For each student, a transaction is created that contains the grades of all courses taken by the student. . As an output, the system generates courses association rules that satisfy the support and confidence constraints. Then the system uses these rules to generate courses recommendations.

Merits of the current system

- (i) Recommendation systems are based on actual user behavior i.e. objective reality. This is the biggest advantage - watching people in their natural environment and making design decisions directly on the results.
- (ii) Recommendation systems are great for discovery.
- (iii) Recommendation systems are effective tools for personalization. We often take recommendations from friends and family because we trust their opinion. They know what we like better than anyone else. This is the sole reason they are good at recommending things.

Most of the organizational maintenance of a site is keeping the navigation system in line with the users' changing needs. With recommendation systems, organizational maintenance is reduced. Based on user activity, the system recommends navigation options to the user. It still takes a designer to decide what type of information should be displayed on what screen.

EXPERIMENT

The traditional system of selecting courses to carry out research work is time consuming, risky and a tedious task, that not only badly affect the performance but the learning experience of a researcher as well. Therefore, choosing appropriate courses in seminal years could help to do research in a better way. This project presents a recommender system that will suggest and guide a learner in selecting the courses as per their requirement. The Hybrid methodology has been used along with ontology to retrieve useful information and make accurate recommendations. Such an approach may be helpful to learners to increase their performance and improve their satisfaction level as well.

Sample Data set taken:

ID	Courses taken
1	Computer networks, cyber security, hardware security
2	C programming , java, python
3	Database, information security,software security
4	Java , python
5	Network optimization, networks
6	C , algorithms, java
7	C and c++, python
8	Cryptography, networks
9	R programming

Apriori algorithm

Apriori Property –

All non-empty subset of frequent itemset must be frequent. The key concept of Apriori algorithm is its anti-monotonicity of support measure.

Steps:

1. Collection of Itemsets
2. Generate K –Item set candidates
3. Support Threshold is compared with the item sets and pruning is performed
4. Final frequent items are identified which satisfies support and confidence threshold values.
5. Stop.

INPUT: S , support where $S = \text{dataset}$, $\text{min_support} = \text{real}$

OUTPUT: Set of Frequent Itemsets

Require: $S \neq \emptyset$, $0 \leq \text{min_support} \leq 1$

```

1: procedure GETFREQUENTITEMSETS
2:   freqSets[ ] ← null
3:   for all Itemsets i in S do
4:     if support ≥ min_support then
5:       freqSets[ ] ← i
6:     end if
7:   end for
8: end procedure
    
```

RESULTS

Set	Support	Confidence	Discovered patterns
{c,C++}	40	80	1254
{java, python}	48	75	2343
{cybersecurity, networks, cryptography}	40	60	3234
{computer networks, cloud computing}	40	85	4233

CONCLUSION FUTURE WORK

The main contribution of this project is a new collaborative recommendation system that employed association rules algorithm to recommend university elective courses to a target student based on what other similar students have taken. The experiments shown that association rule is a desirable tool for making recommendation to a target student. Through our experiments, we noticed the patterns of influence of different parameters on the performance of the system. The confidence and match of a rule have a great impact on the performance, but the highest confidence or match may not be the best choice. By choosing a relatively high confidence or match, we can achieve a better performance.

FUTURE SCOPE

Much work can be performed in the future such as doing comparison between our method and other typical methods. In addition, further experimental evaluation, joining collaborative and content-based recommendations, and applying the new recommendation system in other domains of interest is expected as future work.

REFERENCES

- [1] https://en.wikipedia.org/wiki/Apriori_algorithm
- [2] <https://www.geeksforgeeks.org/apriori-algorithm/>
- [3] Rakesh Agrawal and Ramakrishnan Srikant Fast algorithms for mining association rules. Proceedings of the 20th International Conference on Very Large Data Bases, VLDB, pages 487-499, Santiago, Chile, September 1994.
- [4] The data science behind IP address matching Published by dativa.com, September 6, 2018, retrieved September 7, 2018
- [5] Bayardo Jr, Roberto J. (1998). "Efficiently mining long patterns from databases" (PDF). *ACM SIGMOD Record*. **27** (2).