Knowledge-Based Systems 51 (2013) 1–14

Contents lists available at SciVerse ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Combination of machine learning algorithms for recommendation of courses in E-Learning System based on historical data

Sunita B. Aher [a,*], L.M.R.J. Lobo [b]

[a] Computer Science & Engineering Department, Walchand Institute of Technology, Solapur, Maharashtra, India
[b] Department of Information Technology, Walchand Institute of Technology, Solapur, Maharashtra, India

ARTICLE INFO

Article history:
Received 18 May 2012
Received in revised form 17 April 2013
Accepted 18 April 2013
Available online 25 April 2013

Keywords:
Simple K-means
Apriori
Weka
Moodle
Farthest first
Expectation maximization
Tertius
PredictiveApriori

ABSTRACT

Data mining is the process which is used to analyze the large database to find the useful pattern. Data mining can be used to learn about student's behavior from data collected using the course management system such as Moodle (Modular Object-Oriented Developmental Learning Environment). Here in this paper we show how data mining techniques such as clustering and association rule algorithm is useful in Course Recommendation System which recommends the course to the student based on choice of other students for particular set of courses collected from Moodle. As a result of Course Recommendation System, we can recommend to new student who has recently enrolled for some course e.g. Operating System, the new course to be opted e.g. Distributed System. Our approach uses combination of clustering technique – Simple K-means and association rule algorithm – Apriori and finds the result. These results were compared with the results of open source data mining tool-Weka. The result obtained using combined approach matches with real world interdependencies among the courses. Other combinations of clustering and association rule algorithms are also discussed here to select the best combination. This Course Recommendation System could help in building intelligent recommender system. This approach of recommending courses to new students can be immensely be useful in "MOOC (Massively Open Online Courses)".

1. Introduction

Data mining is the process of discovering interesting knowledge from the large database. Data mining is used in banking, telecommunication industry, DNA analysis, etc. Data mining involves tasks such as outlier detection, association rule learning, classification, clustering, regression and summarization [1].

Recommender systems have obtained success in E-commerce [31]. Personalized product recommendations assist customers in finding products they would like to purchase by producing a list of recommended products [25]. The Course Recommendation System in e-learning is a system that suggests the best combination of subjects in which the students are interested.

Data mining and machine learning techniques that could be used to enhance web-based learning environments for the educator to better evaluate the leaning process [32]. Data mining can be used to learn about student's behavior in Educational System. To learn about this behavior, we use data collected from the Learning Management System (LMS). Here we consider LMS such as Moodle (Modular Object-Oriented Developmental Learning Environment) which allows the instructor to create courses, take online test, etc.

Here in this Course Recommendation System, we recommend student, a course in which he/she might be interested e.g. if a student opts to learn a course like System Programming then he would be keen to learn the course like Compiler Construction.

2. Literature review

Numerous studies have been done in E-Learning System which focused on behavior of student. They deal with different data mining techniques for analyzing students' behavior.

Castro et al. [2] provided an up-to-date snapshot of the current state of research and applications of data mining methods in e-learning. It provided the taxonomy of e-learning problems to which the data Mining techniques have been applied including, for instance: Students' classification based on their learning performance; detection of irregular learning behaviors; E-Learning System navigation and interaction optimization; clustering according to similar E-Learning System usage; and systems' adaptability to students' requirements and capacities.

Romero et al. [3] described the full process for mining e-learning data step by step as well as how to apply the main data mining techniques used, such as statistics, visualization, classification,

* Corresponding author.
E-mail addresses: sunita_aher@yahoo.com (S.B. Aher), headitwit@gmail.com (L.M.R.J. Lobo).

0950-7051/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.knosys.2013.04.015

clustering and association rule mining of MOODLE data. They installed GISMO into the MOODLE system which is a graphical interactive student monitoring and tracking system tool that extracts tracking data from MOODLE.

Seki et al. [4] developed an integrated e-learning environment which integrated learning history and learning content information to control each learner's learning. They have developed an LMS based on Learning Ecological Model which is a model of learning environment focusing on learning content, learning objective, and learning style. By analyzing access log and report submission log, they obtained suggestions for improvement such as restriction of browsing period to make students access constantly, fragmentation of report submission deadline for students not keeping report assignment too.

Zaïane and Luo [5] exploited the existence of web access logs and advanced data mining techniques to extract useful patterns that can help educators and web masters evaluate and interpret on-line course activities in order to assess the learning process, track students actions and measure web course structure effectiveness.

Carmona et al. [6] proposed how to use adaptive machine learning algorithms to learn about the student's preferences over time. First they used all the background knowledge available about a particular student to build an initial decision model based on learning styles. This model can then be fine-tuned with the data generated by the student's interactions with the system in order to reflect more accurately his/her current preferences.

Hamalainen et al. [7] was designed and implemented Data Mining System (DMS) to analyze the study records of two programming courses in a distance curriculum of Computer Science. Various data mining schemes, including linear regression and probabilistic models, were applied to describe and predict student performance.

Itmaz and Megias [8] discussed the recommendation system ability in learning management system/course management system to support student need as well as it discussed the suitability of every recommendation system approach to recommend the learning objects. This paper studies and evaluates the ability to use recommender system in learning management system as well as designing a new recommendation system algorithm to recommend the list of suitable courses.

Various recommendation techniques, like data mining, agents and reasoning, have been developed and applied into recommender systems [24,25,28–30].

Verbert et al. [18] try to assess the degree to which current work in Technology Enhanced Learning (TEL) recommender systems has achieved this, as well as outline areas in which further work is needed. Lu [19] proposes a framework of a personalized learning recommender system, which aims to help students find learning materials they would need to read. Zaïane [20] suggested the use of web mining techniques to build recommender system that could recommend on-line learning activities or shortcuts in a course web site based on learners' access history to improve course material navigation as well as assist the online learning process.

Khribi et al. [21] described an automatic personalization approach aiming to provide online automatic recommendations for active learners without requiring their explicit feedback. In the paper [22], García et al. describes a collaborative educational data mining tool based on association rule mining for the ongoing improvement of e-learning courses and allowing teachers with similar course profiles to share and score the discovered information.

In paper [23], Tang and McCalla propose an evolving web-based learning system which can adapt itself not only to its users, but also to the open Web.

Changchien and Lu [24] developed a mining association rules procedure from a database to support on-line recommendation. Kim et al. [26] introduce a personalized recommendation procedure by which they can get further recommendation effectiveness when applied to Internet shopping malls. Cho et al. [27] suggests a personalized recommendation methodology by which they are able to get further effectiveness and quality of recommendations when applied to an Internet shopping mall.

3. Framework for Course Recommendation System

In this Course Recommendation System, we consider 13 course categories and 82 courses that are related to two branches i.e. Computer Science & Engineering (CSE) and Information Technology (IT) to collect data from students. After collecting the data we consider only those students who enroll for at least 5 courses

Table 1
Course categories and courses.

Sr. No.	Course category, course name and code
1	**Basic Science (BS)** 1. Discrete Mathematical Structure (DMS)
2	**Humanities & Social Science (HSS)** 1. Communication Skills (COMM)
3	**Hardware Design & Engineering (HDE)** 1. Switching Theory And Logic Design (STLD) 2. Microprocessor (MP) 3. Computer Organization (CO) 4. Advanced Computer Architecture (ACA) 5. Mobile Computing (MC)
4	**Theoretical Computer Science (TCS)** 1. Data Structures –I (DS-I) 2. Data Structures –II (DS-II) 3. Design & Analysis of Algorithm (DAA) 4. Formal System And Automata (FSA)
5	**System software (SS)** 1. System Programming (SP) 2. Operating System – I (OS-I) 3. Operating System – II (UNIX) (OS-II)
6	**Networks (NT)** 1. Computer Networks – I (CN-I) 2. Computer Networks – II (CN-II) 3. Network Security (NS)
7	**Database (DB)** 1. Database Engineering (DBE) 2. Oracle (SQL)
8	**Software Engineering &Principles (SEP)** 1. Software Engineering (SE) 2. Software Testing & Quality Assurance (STQA)
9	**Image Processing, Graphics & Artificial Intelligence Application (IP&AIP)** 1. Computer Graphics (CG) 2. Artificial Intelligence (AI) 3. Human Computer Interfaces (HCI)
10	**Web Technology (WT)** 1. Hyper Text Markup Language (HTML) 2. JavaScript (JS) 3. Java Server Pages (JSP) 4. ASP.NET (ASP.NET)
11	**Principles of Programming Language (PPL)** 1. Visual Basic (VB) 2. C-Programming (CP) 3. Advanced C (AC) 4. Object Oriented Design & Programming (OODP) 5. Java Programming (JP) 6. Advanced Java (AJ) 7. C-Sharp (CS)
12	**Information Retrieval & Extraction (IRE)** 1. Data Mining (DM)

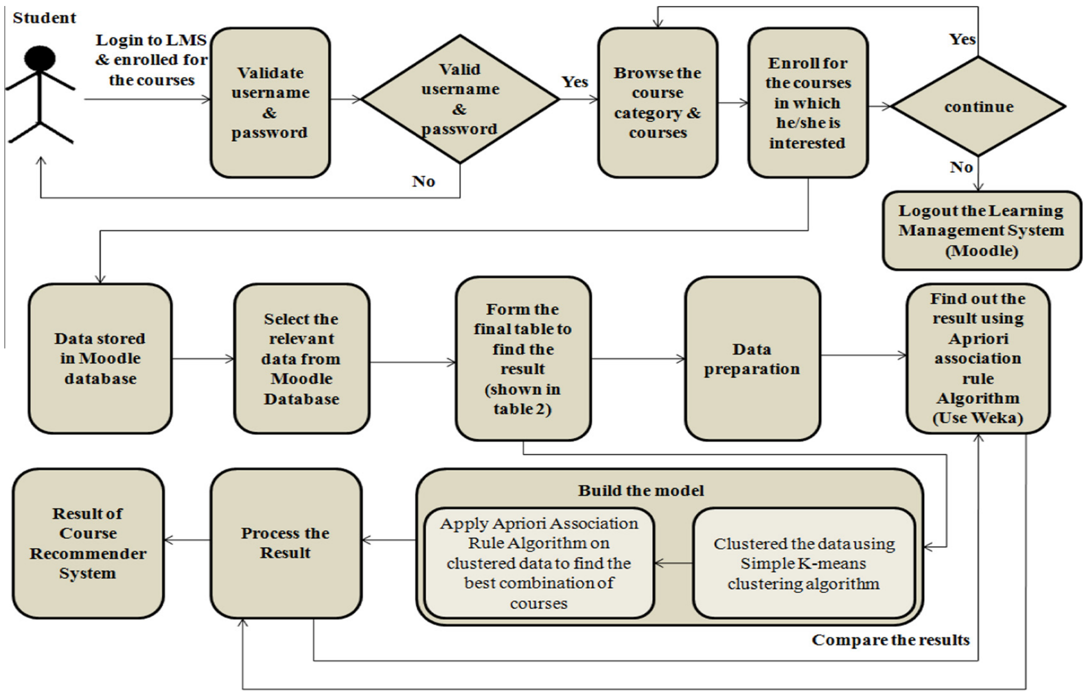

Fig. 1. Framework for Course Recommendation System in e-learning using clustering and association rule algorithm.

Table 2
Final table structure to find the results of Course Recommendation System.

Roll no. of student	Course												
	DMS	CS	STLD	MP	...	DBE	SQL	...	OODP	JP	AJ	CS	DM
Becse01	Yes	Yes	Yes	No	...	Yes	Yes	...	Yes	Yes	No	No	No
Becse0	No	No	No	No	...	Yes	Yes	...	Yes	Yes	No	No	No
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
Beit01	Yes	No	No	No	...	Yes	No	...	No	No	No	Yes	Yes
Beit02	No	Yes	Yes	Yes	...	No	Yes	...	Yes	Yes	No	No	No
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
Tecse01	No	No	No	No	...	Yes	Yes	...	No	No	No	No	No
Tecse02	Yes	Yes	Yes	Yes	...	No	No	...	Yes	No	No	No	No
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
Teit01	Yes	No	No	No	...	Yes	Yes	...	Yes	Yes	No	Yes	Yes
Teit02	No	Yes	Yes	Yes	...	Yes	No	...	No	No	No	Yes	Yes
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
Secse101	Yes	Yes	Yes	Yes	...	No	Yes	...	Yes	No	No	No	No
Secse102	Yes	Yes	Yes	Yes	...	No	No	...	Yes	Yes	Yes	No	No
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
Seit01	No	Yes	Yes	Yes	...	No	No	...	Yes	Yes	Yes	No	No
Seit02	Yes	Yes	Yes	Yes	...	No	Yes	...	Yes	No	No	No	No
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
Secse201	Yes	No	No	No	...	No	Yes	...	Yes	No	No	No	No
Secse202	Yes	Yes	Yes	Yes	...	No	No	...	Yes	Yes	Yes	No	No
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

and only those courses which have been opted by more than 100 students. As we consider 83 courses, some courses may not be opted by students. So after forming the final table, we consider only those courses which have been opted by at least 100 students. So finally we consider 12 course categories and 36 courses which are shown in Table 1.

The framework for Course Recommendation System is explained in [9]. The framework for Course Recommendation System using combination of Simple K-means clustering and Apriori association rule algorithm is given in Fig. 1. After logging into Moodle system, student's username and password is validated. If it is valid then student browses the course categories and courses. After browsing these courses, student will enroll for those set of courses in which he/she is interested. Here student should enroll for at least five courses otherwise the data of that student will not be considered while forming the final table to find the result. Enrollment for set of particular courses information is stored in Moodle database. So we select the relevant data from Moodle database and form the final table shown in Table 2 to find the result of Course Recommendation System. Here we are using open source data mining tool Weka [10] to find the result.

Weka is a collection of machine learning algorithms for data mining tasks and the algorithms can either be applied directly to a dataset or called from your own Java code [11]. The result of application of Apriori association rule algorithm in Table 2 gives negative association rules. As we are recommending the course to the student, we require all positive association rules. So we prepare Table 2. In Fig. 1, Preparation of data step is explained in [16]. After this data preparation step, if we apply Apriori association rule algorithm on table obtained as a result of this stage, then we get positive association rules.

Next step is building the model, in this step first we apply Simple K-means clustering algorithm in Table 2. After clustering the data in Table 2, we apply Apriori association rule algorithm on each cluster to find the result.

Finally we compare the result of this combined approach with the result using Apriori association rule algorithm, process the result and display the result of Course Recommendation System. Also we show how the result of combined approach differs from the result using Apriori association rule algorithm.

4. Combined approach using Simple K-means clustering and Apriori association rule algorithm

4.1. Association rule algorithm for Course Recommendation System

Association rules are used to show the relationship between data items. Mining association rules allows finding rules of the form: If antecedent then (likely) consequent where antecedent and consequent are itemsets which are sets of one or more items. Apriori Association rule is used to mine the frequent patterns in database.

Table 3 show the comparison among three association rule algorithms i.e. Tertius association rule algorithm, PredictiveApriori association rule algorithm and Apriori association rule algorithm to choose the best association rule. Here Apriori association rule algorithm is considered for this Course Recommendation System as this algorithm gives the positive association rule i.e. association rule containing "yes". The rules obtained using Apriori association rule algorithm matches with real world interdependencies among courses e.g. "Data_Structure_II=yes → Data_Structure_I=yes", the meaning of this association rule is "If student is interested in Data structure-I then he would like the course Data Structure-II".

Other association algorithm such as Tertius or PredictiveApriori gives negative association rules i.e. rules in the form of "yes" and "no" which is explained in rows numbered one and two of Table 3 e.g. "Operating_System_I=no Computer_Network_I=no → Data_Structure_I=yes", the meaning of the rule is "If the student is interested in Data Structure-I then he does not like the course like Operating System and Computer Network".

As this is the Course Recommendation System, we need only positive association rules i.e. rules in the form of "yes" only, hence we consider Apriori association rule algorithm for recommending the course to the students [12].

4.2. Clustering algorithm for Course Recommendation System

Clustering is finding groups of objects such that the objects in one group will be similar to one another and different from the objects in another group.

Table 3
Comparison of association rule algorithm.

Sr. No.	Result of association rule algorithms using open source data mining tool WEKA
1	**Tertius Association Rule** Best rules found: 1. Computer Network-I = yes and Java Programming = yes Operating System-I = yes or Visual Basic = no 2. Data Structure-I = yes and Computer Network-I = no and Visual Basic = yes Operating System-I = no 3. Data Structure-I = yes and Operating System-I = no Computer Network-I = no or Java Programming = no 4. Data Structure-II = yes and Operating System-I = no Computer Network-I = no or Java Programming = no 5. Computer Network-I = yes and Java Programming = yes Operating System-I = yes or C-Programming = no 6. Computer Network-I = yes and Java Programming = yes Operating System-I = yes or Switching Theory & Logic Design = no 7. Visual Basic = no Operating System-I = yes or Computer Network-I = yes 8. Computer Network-I=yes and Java Programming=yes Operating System-I = yes 9. Data Structure-II = yes and Visual Basic = yes Operating System-I = no or C-Programming = yes 10. Data Structure-I = yes and Visual Basic = yes Operating System-I = no
2	**PredictiveApriori Association Rule** Best rule found: 1. Data_Structure_II =yes Operating_System_I =no Java_Programming =yes Data_Structure_I =yes 2. Operating_System_I=no Computer_Network_I =no Data_Structure_I =yes 3. Data_Structure_II =yes Operating_System_I =no Data_Structure_I =yes 4. Switching_Theory_&_Logic_Design =no Operating_System_I =no Data_Structure_I =yes 5. Switching_Theory_&_Logic_Design =no Data_Structure_II =no Java_Programming =yes 6. Operating_System_I =no Visual_Basic =no Data_Structure_I =yes 7. Operating_System_I =no Computer_Network_I =no Visual_Basic =yes 8. Switching_Theory_&_Logic_Design = yes Operating_System_I =no C_Programming =no Data_Structure_I =yes 9. Operating_System_I =no Computer_Network_I =no Java_Programming =no Switching_Theory_&_Logic_Design =yes 10. Operating_System_I =no Computer_Network_I =no Visual_Basic =yes Java_Programming =no C_Programming =yes
3	**Apriori Association Rule Algorithm** **1.Minimum support: 0.3** **2.Minimum metric <confidence>:0.9** **3. Number of cycles performed: 14**

Best rules found:
1. Data_Structure_II =yes Data_Structure_I =yes conf:(0.93)
2. Data_Structure_II =yes Java_Programming =yes Data_Structure_I =yes conf:(0.93)
3. Data_Structure_II =yes Visual_Basic =yes Data_Structure_I = yes conf:(0.93)
4. Data_Structure_II =yes Computer_Network_II=yes C_Programming =yes Data_Structure_I =yes conf:(0.93)
5. Switching_Theory_&_Logic_Design =yes Data_Structure_II =yes C_Programming =yes Data_Structure_I =yes conf:(0.93)
6. Data_Structure_II=yes C_Programming= yes Data_Structure_I=yes conf:(0.92)
7. Data_Structure_II =yes Visual_Basic =yes C_Programming =yes Data_Structure_I =yes conf:(0.92)
8. Switching_Theory_&_Logic_Design =yes Data_Structure_II =yes Data_Structure_I =yes conf:(0.92)
9. Data_Structure_II =yes Computer_Network_I =yes Data_Structure_I =yes conf:(0.92)
10. Data_Structure_II =yes C_Programming =yes Java_Programming =yes Data_Structure_I = yes conf:(0.92)

1.Minimum support: 0.4
2.Minimum metric <confidence>: 0.9
3.Number of cycles performed: 12

Best rules found:
1. Data_Structure_II =yes Data_Structure_I =yes conf:(0.93)
2. Data_Structure_II =yes Java_Programming =yes Data_Structure_I =yes conf:(0.93)
3. Data_Structure_II =yes Visual_Basic =yes Data_Structure_I =yes conf:(0.93)
4. Data_Structure_II =yes C_Programming =yes Data_Structure_I =yes conf:(0.92)
5. Switching_Theory_&_Logic_Design =yes Data_Structure_II =yes Data_Structure_I =yes conf:(0.92)
6. Data_Structure_II =yes Computer_Network_I =yes Data_Structure_I =yes conf:(0.92)
7. Data_Structure_II =yes Operating_System_I =yes Data_Structure_I =yes conf:(0.9)

1.Minimum support: 0.5
2.Minimum metric <confidence>: 0.9
3.Number of cycles performed: 10

Best rules found:
1. Data_Structure_II =yes Data_Structure_I =yes conf:(0.93)
2. Data_Structure_II =yes C_Programming =yes Data_Structure_I =yes conf:(0.92)

1.Minimum support: 0.6
2.Minimum metric <confidence>: 0.9
3.Number of cycles performed: 8

Best rules found:
1. Data_Structure_II =yes Data_Structure_I =yesconf:(0.93)

Simple K-means algorithm is a clustering algorithm in which items are moved among the set of cluster until required set is reached. This algorithm is used to classify the data set, provided the number of cluster is given in prior. This algorithm is iterative in nature [1].

Farthest First is a variant of K Means that places each cluster center in turn at the point furthermost from the existing cluster center. This point must lie within the data area. This greatly speeds up the clustering in most of the cases since less reassignment.

In statistics, an expectation–maximization (EM) algorithm is an iterative method for finding maximum likelihood or maximum a posteriori (MAP) estimates of parameters in statistical models, where the model depends on unobserved latent variables. The EM iteration alternates between performing an expectation (E) step, which computes the expectation of the log-likelihood evaluated using the current estimate for the parameters, and maximization (M) step, which computes parameters maximizing the expected log-likelihood found on the E step. These parameter-estimates are then used to determine the distribution of the latent variables in the next E step [13]. EM assigns a probability distribution to each instance which indicates the probability of it belonging to each of the clusters. EM can decide how many clusters to create by cross validation, or you may specify Apriori how many clusters to generate.

Table 4 shows the comparison among three clustering algorithms i.e. Simple K-means clustering, Farthest First clustering and Expectation Maximization clustering algorithm after application of Apriori association rule algorithm on clustered data to select the best clustering algorithm.

The result after application of Apriori association rule algorithm on clustered data using Simple K-means clustering algorithm matches with the real world interdependencies among the courses as shown in row numbered one of Table 4.

The rules obtained after application of Apriori association rule algorithm on clustered data using Farthest First and Expectation Maximization clustering algorithm do not much adhere to real world choices of students. From rows numbered two and three of Table 4, we find that in real time student did not opt for such combination of courses as in the rules obtained by Expectation Maximization clustering algorithm ‘‘Microprocessor=yes ==> Data_Structure_I=yes, Microprocessor=yes ==> Operating_System_I=yes, Microprocessor=yes ==> Java_Programming=yes “ and rules by Farthest First clustering algorithm ‘‘Visual_Basic=yes ==> Java_Programming=yes, Microprocessor=yes Operating_System_I=yes

Table 4
Result comparison after application of Apriori association rule algorithm on the clustering algorithm.

Sr. No.	Result of combination of Clustering & Association Rule Algorithm
1	**After application of Apriori association rule algorithm on the clustered data using Simple K-means clustering algorithm** **Minimum support: 0.85 (27 instances)** **Minimum metric <confidence>: 0.9** **Number of cycles performed: 3** **Best Rules found:** **1. Operating_System_I=yes ==> Java_Programming=yes** **2. C_Programming=yes ==> Java_Programming=yes** **3. Operating_System_I=yes C_Programming=yes ==> Java_Programming=yes** **4.Object_Oriented_Design_&_Programming=yes ==> Java_Programming=yes** **5. Java_Programming=yes ==> Operating_System_I=yes** **6. Java_Programming=yes ==> C_Programming=yes** **7. C_Programming=yes ==> Operating_System_I=yes** **8. Operating_System_I=yes ==> C_Programming=yes** **9. C_Programming=yes Java_Programming=yes ==> Operating_System_I=yes** **10. Operating_System_I=yes Java_Programming=yes ==> C_Programming=yes**
2	**After application of Apriori association rule algorithm on the clustered data using Farthest First clustering algorithm** Minimum support: 0.75 (30 instances) Minimum metric <confidence>: 0.9 Number of cycles performed: 5 Best Rules found: 1.Operating_System_I=yes ==> Java_Programming=yes 2. C_Programming=yes ==> Java_Programming=yes 3. Advanced_Java=yes ==> Java_Programming=yes 4. Visual_Basic=yes ==> Java_Programming=yes 5. Operating_System_I=yes C_Programming=yes ==> Java_Programming=yes 6. Operating_System_I=yes Advanced_Java=yes ==> Java_Programming=yes 7. Microprocessor=yes Operating_System_I=yes ==> Java_Programming=yes 8. Operating_System_I=yes Visual_Basic=yes ==> Java_Programming=yes 9. Visual_Basic=yes C_Programming=yes ==> Java_Programming=yes 10. Data_Structure_II=yes ==> Java_Programming=yes
3	**After application of Apriori association rule algorithm on the clustered data using Expectation Maximization clustering algorithm** Minimum support: 0.95 (14 instances) Minimum metric <confidence>: 0.9 Number of cycles performed: 1 Best Rules found: 1. Java_Programming=yes ==> Operating_System_I=yes 2.Operating_System_I=yes ==> Java_Programming=yes 3. Data_Structure_I=yes ==> Microprocessor=yes 4. Microprocessor=yes ==> Data_Structure_I=yes 5. Microprocessor=yes ==> Operating_System_I=yes 6. Microprocessor=yes ==> Java_Programming=yes 7. Data_Structure_I=yes ==> Operating_System_I=yes 8. Data_Structure_I=yes ==> Java_Programming=yes 9. Data_Structure_II=yes ==> Operating_System_I=yes 10.Data_Structure_II=yes ==> Java_Programming=yes

==> Java_Programming=yes, Operating_System_I=yes Visual_Basic=yes ==> Java_Programming=yes". As stated in [14], Simple K-means performs better than Farthest First clustering algorithm.

Hence we consider Simple K-means algorithm for Course Recommendation System which is explained in [15].

4.3. Best combination of clustering and association rule algorithm

From Table 4, the combined approach using Simple K-means clustering and Apriori association rule algorithm is better than the combined approach using Farthest First clustering algorithm and Apriori association rule algorithm.

Also Simple K-means performs better than Farthest First clustering algorithm as stated in [14].

We also take into account support, number of cycles performed and generated set of large itemsets to select the best combination clustering and association rule algorithm.

EM clustering method needs more time than other clustering algorithm. Also the association rules obtained using combination of Expectation Maximization clustering and Apriori association rule algorithm and using combination of Farthest First clustering and Apriori association rule algorithm does not match with real world choices of students as in row numbered 3 of Table 4. So the combined approach using Expectation Maximization clustering and Apriori association rule algorithm is not suitable to Course Recommendation System.

Table 5 shows the comparison for selecting the best combination of clustering and association rule algorithm as combination of Simple K-means clustering and Apriori association rule algorithm.

4.4. Combined algorithm using Simple K-means clustering and Apriori association rule algorithm

The algorithm for Simple K-means clustering and Apriori association rule algorithm is given in Fig. 2. The algorithm and flowchart for Course Recommendation System using combination of

Table 5
Comparison for selecting the best combination of clustering and association rule algorithm.

Parameter considered	Combined Approach using Simple K-means clustering and Apriori association rule algorithm	Combination Farthest First clustering and Apriori association rule algorithm	Combination Expectation Maximization clustering and Apriori association rule algorithm
Support	0.85	0.75	0.95
No. of Rules	10	10	10
No. of cycles performed	3	5	1
No. of clusters	5	2	11
Time required	0.08 s	0.02 s	471.67 s
Generated sets of Large Itemset	• Size of set of large itemsets L(1): 4 • Size of set of large itemsets L(2): 4 • Size of set of large itemsets L(3): 1	• Size of set of large itemsets L(1): 12 • Size of set of large itemsets L(2): 13 • Size of set of large itemsets L(3): 5	• Size of set of large itemsets L(1): 7 • Size of set of large itemsets L(2): 12 • Size of set of large itemsets L(3): 7 • Size of set of large itemsets L(4): 1

Algorithm: Simple K-means clustering algorithm
Input:
Set of Elements or Database of transaction
D= $\{t_1, t_2, t_3, \ldots, t_n\}$
Number of required Cluster k
Output:
Set of Cluster K
Method:
Make initial guesses for the means $\mathbf{m_1}, \mathbf{m_2}, \ldots, \mathbf{m_k}$;
Repeat
Assign each element t_i to the cluster having the Closest mean.
Calculate the new mean for each cluster.
Until there are no changes in any mean
	Algorithm: Apriori Association Rule Algorithm
	Input :
	Database of Transactions D=$\{t_1,t_2, \ldots, t_n\}$
	Set if Items I= $\{I_1, I_2, \ldots I_k\}$
	Frequent (Large) Itemset L
	Support,
	Confidence.
	Output :
	Association Rule satisfying Support & Confidence
	Method:
	C_1 = Itemsets of size one in I;
	Determine all large itemsets of size 1, L_1;
	i = 1;
	Repeat
	i = i + 1;
	C_i = Apriori-Gen(L_{i-1});
	Apriori-Gen(L_{i-1})
	•Generate candidates of size i+1 from large itemsets of size i.
	•Join large itemsets of size i if they agree on i-1.
	•Prune candidates who have subsets that are not large.
	Count C_i to determine L_i;
	until no more large itemsets found;

Fig. 2. Simple K-means clustering and Apriori association rule algorithm.

Simple K-means clustering and Apriori association rule algorithm is given in Figs. 3 and 4.

First we apply Simple K-means clustering algorithm on ARFF (Attribute Relation File Format) file for Table 2 which is formed using data collected from Moodle database. By providing number of cluster, the ARFF file is divided into that many numbers of clusters. After clustering the data, we apply Apriori association rule algorithm on each cluster. We select the result of that cluster for recommending the course which gives the correct result. Correct result means, that all association rules are positive containing "yes" as we are recommending the course to the student and ignore the result of remaining clusters as association rules of these clusters contains negative association rules having "no".

4.5. Comparative study of combined approach (using Simple K-means clustering and Apriori association rule algorithm) and Apriori association rule

Here we consider the comparative study of combined approach using Simple K-means clustering and Apriori association rule algorithm and Apriori association rule.

From the results obtained after application of Apriori association rule algorithm as shown in row numbered three of Table 3, we see that, as we increase the support then we get the refined rule but the number of rules, we get, are less. For support value of 0.6 we get only one association rule.

1. Enter name of ARFF file, number of cluster (n) & seed value for Simple K-means clustering algorithm.
2. Apply Simple K-means clustering algorithm on mentioned ARFF file.
3. Separate number of cluster n into separate file m (m=n).
4. For I =1 to m
   Apply Apriori association rule algorithm on that file.
5. If result of none of file contains correct association rule algorithm
   then
   go to step 1
   else
   go to step 6.
   [Correct association rules means all rules having "yes" only.]
6. Display the result of correct file containing valid association rules for Course Recommendation System

Fig. 3. Algorithm using combination of Simple K-means clustering and Apriori association rule algorithm.

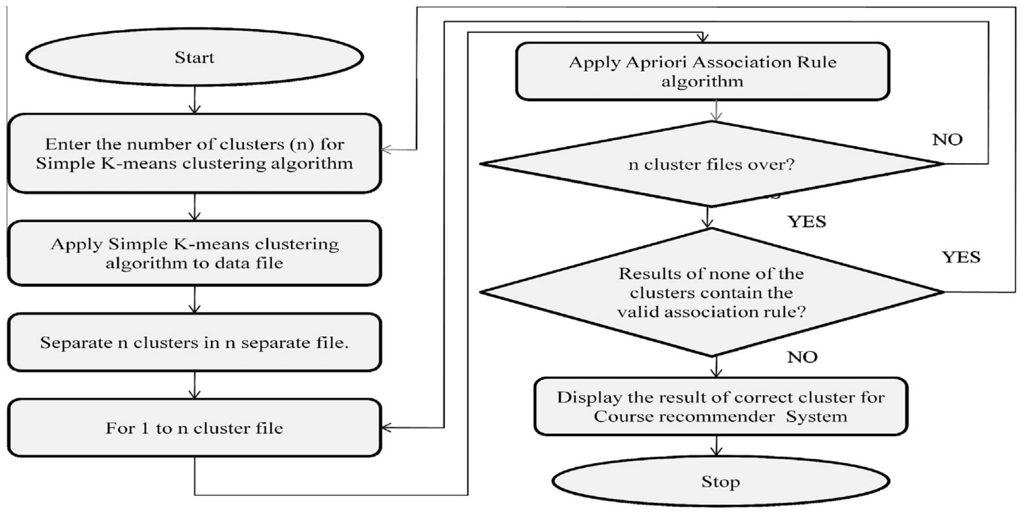

Fig. 4. Flowchart for combined approach using Simple K-means clustering and Apriori association rule algorithm.

If we use the combined approach using combination of Simple K-means clustering and Apriori association rule algorithm then we get ten association rules for support value of 0.85 which is shown in row numbered 3 of Table 6. As the number of association rules are more, we can recommend various combination of courses to the students which is not the case with the results using Apriori association rule algorithm.

Also with the combined approach, there is no need to prepare the data which is explained in paper [16] but if we consider the result using Apriori association rule algorithm then there is need to prepare the data.

Table and Graph showing the comparative study of combined approach and Apriori association rule algorithm is shown in Table 6 and Fig. 5 respectively.

4.6. Time and space complexity of combined approach

4.6.1. Time complexity

Time complexity of Apriori is $O(d^2n)$, where d is number of items and n is number of records [17]. Time complexity of Simple K-means is $O(tkn)$, where t is number of iteration, k is number of cluster and n is number of records [1].

Time complexity of algorithm using combination of Simple K-means clustering and Apriori association rule algorithm is shown in Fig. 6.

So the time complexity of algorithm using combined approach of Simple K-means clustering and Apriori association rule algorithm is $O(d^2n) + C_{SA}$.

Table 6
Comparative study of combined approach and Apriori association rule algorithm.

Algorithm considered	Parameter considered			
	Support	No. of rules	No. of cycles performed	Generated sets of large itemset
Apriori association rule algorithm	0.3	10	14	• Size of set of large itemsets L(1): 12 • Size of set of large itemsets L(2): 29 • Size of set of large itemsets L(3): 48 • Size of set of large itemsets L(4): 17
Apriori association rule algorithm	0.4	7	12	• Size of set of large itemsets L(1): 8 • Size of set of large itemsets L(2): 28 • Size of set of large itemsets L(3): 21
Apriori association rule algorithm	0.5	2	10	• Size of set of large itemsets L(1): 8 • Size of set of large itemsets L(2): 17 • Size of set of large itemsets L(3): 2
Apriori association rule algorithm	0.6	1	8	• Size of set of large itemsets L(1): 8 • Size of set of large itemsets L(2): 6
Combined Approach using Simple K-means clustering and Apriori association rule	0.85	10	3	• Size of set of large itemsets L(1): 4 • Size of set of large itemsets L(2): 4 • Size of set of large itemsets L(3): 1

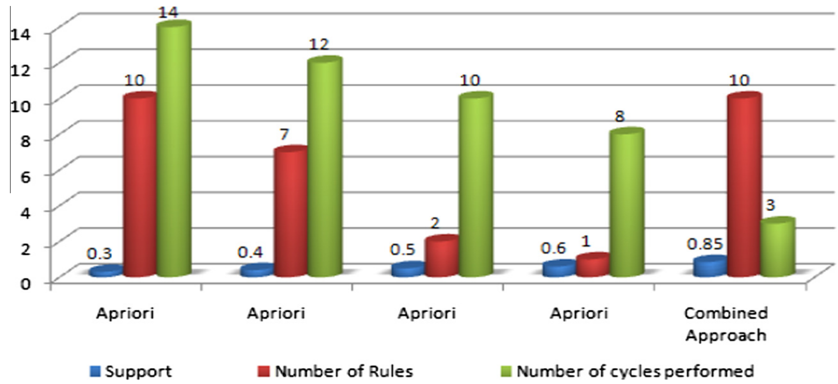

Fig. 5. Graph showing the Comparative study of combined approach and Apriori association rule algorithm.

$$\begin{aligned}\text{Time complexity of combined approach} &= \max(\text{Time complexity of Simple K-means clustering algorithm},\\ &\qquad \text{Time complexity of Apriori association rule algorithm})\\ &\quad + C_{SA}\\ &= \max(O(tkn), O(d^2 n)) + C_{SA}\\ &= O(d^2 n) + C_{SA} \text{ since } tk \ll d^2,\end{aligned}$$

where C_{SA} denotes any constants.

Fig. 6. Time complexity of algorithm using combination of Simple K-means clustering and Apriori association rule algorithm.

$$\begin{aligned}\text{Space complexity of combined approach} &= \text{space complexity of Simple K-means clustering}\\ &\quad + \text{space complexity of Apriori association rule algorithm}\\ &= O(n + C_A) + O(n + C_S)\\ &= O(n + C_A + C_S)\end{aligned}$$

Fig. 7. Space complexity of algorithm using combination of Simple K-means clustering and Apriori association rule algorithm.

Table 7
Result using data mining tool Weka and combined approach.

Sr. No.	Results
1	**Before data preparation stage** 36 courses considered in Table 1. Minimum support: 0.6 (418 instances) Minimum metric <confidence>: 0.9 Number of cycles performed: 8 Best rules found: 1. Oracle=no Asp_Dot_Net=no C_Sharp=no conf:(0.94) 2. Design_&_Anaylisis_Of_Algorithm=no System_Programming=no Finite_State_Automata=no conf:(0.93) 3. Asp_Dot_Net=no Advanced_C=no C_Sharp=no conf:(0.92) 4. Asp_Dot_Net=no Data_Mining=no C_Sharp=no conf:(0.92) 5. Computer_Organisation=no System_Programming=no Finite_State_Automata=no conf:(0.92) 6. Design_&_Anaylisis_Of_Algorithm=no Oracle=no C_Sharp=no conf:(0.92) 7. Oracle=no Java_Server_Pages =no C_Sharp=no conf:(0.92) 8. System_Programming=no Java_Server_Pages=no Finite_State_Automata=no conf:(0.92) 9. Oracle=no Javascript=no C_Sharp=no conf:(0.92) 10.Oracle=no Data_Mining=no C_Sharp=no conf:(0.92)
2	**After data preparation stage** Courses considered: Switching Theory & Logic Design, Data Structure-I, Data Structure-II, Operating System-I, Computer Network-I, Visual Basic, C-Programming, Java Programming Apriori Association Rule Algorithm 1.Minimum support: 0.3 2.Minimum metric <confidence>:0.9 3. Number of cycles performed: 14 Best rules found: 1. Data_Structure_II =yes Data_Structure_I =yes conf:(0.93) 2. Data_Structure_II =yes Java_Programming =yes Data_Structure_I =yes conf:(0.93)

(continued on next page)

	3. Data_Structure_II =yes Visual_Basic =yes Data_Structure_I = yes conf:(0.93) 4. Data_Structure_II =yes Computer_Network_II=yes C_Programming =yes Data_Structure_I =yes conf:(0.93) 5. Switching_Theory_&_Logic_Design =yes Data_Structure_II =yes C_Programming =yes Data_Structure_I =yes conf:(0.93) 6. Data_Structure_II=yes C_Programming= yes Data_Structure_I=yes conf:(0.92) 7. Data_Structure_II =yes Visual_Basic =yes C_Programming =yes Data_Structure_I =yes conf:(0.92) 8. Switching_Theory_&_Logic_Design =yes Data_Structure_II =yes Data_Structure_I =yes conf:(0.92) 9. Data_Structure_II =yes Computer_Network_I =yes Data_Structure_I =yes conf:(0.92) 10. Data_Structure_II =yes C_Programming =yes Java_Programming =yes Data_Structure_I = yes conf:(0.92) 1.Minimum support: 0.4 2.Minimum metric <confidence>: 0.9 3.Number of cycles performed: 12 Best rules found: 1. Data_Structure_II =yes Data_Structure_I =yes conf:(0.93) 2. Data_Structure_II =yes Java_Programming =yes Data_Structure_I =yes conf:(0.93) 3. Data_Structure_II =yes Visual_Basic =yes Data_Structure_I =yes conf:(0.93) 4. Data_Structure_II =yes C_Programming =yes Data_Structure_I =yes conf:(0.92) 5. Switching_Theory_&_Logic_Design =yes Data_Structure_II =yes Data_Structure_I =yes conf:(0.92) 6. Data_Structure_II =yes Computer_Network_I =yes Data_Structure_I =yes conf:(0.92) 7. Data_Structure_II =yes Operating_System_I =yes Data_Structure_I =yes conf:(0.9) 1.Minimum support: 0.5 2.Minimum metric <confidence>: 0.9 3.Number of cycles performed: 10 Best rules found: 1. Data_Structure_II =yes Data_Structure_I =yes conf:(0.93) 2. Data_Structure_II =yes C_Programming =yes Data_Structure_I =yes conf:(0.92) 1.Minimum support: 0.6 2.Minimum metric <confidence>: 0.9 3.Number of cycles performed: 8 Best rules found: 1. Data_Structure_II =yes Data_Structure_I =yes conf:(0.93)
3	**Result of application of combined approach using Simple K-means clustering Apriori association rule algorithm on data obtained from Moodle database** Courses considered:36 courses considered in Table 1 Result after application of combination of Clustering & Association rule algorithm Enter name of ARFF file (with .arff):BPm2m.arff Number of Cluster:5 Enter seed value:100 ****SIMPLE K-MEANS CLUSTERING ALGORITHM**** Number of iterations: 6 Within cluster sum of squared errors: 5544.0 Missing values globally replaced with mean/mode Cluster centroids: Cluster# Attribute Full Data 0 1 2 3 4 (697) (333) (155) (32) (96) (81) == Discrete_Mathematical_Structure no no no yes no yes

Communication_Skills	no	no	no	no	no	no
Switching_Theory_&_Logic_Design	no	no	yes	yes	no	yes
Microprocessor	no	no	yes	yes	no	no
Computer_Organisation	no	no	no	yes	no	no
Advanced_Computer_Architecture	no	no	no	no	no	no
Mobile_Computing	no	no	no	yes	no	no
Data_Structure_I	yes	no	yes	yes	yes	yes
Data_Structure_II	no	no	yes	yes	no	no
Design_&_Anaylisis_Of_Algorithm	no	no	no	yes	no	no
Formal_System_And_Automata	no	no	no	yes	no	no
System_Programming	no	no	no	yes	no	no
Operating_System_I	no	no	yes	yes	no	no
Operating_System_II	no	no	no	yes	no	no
Computer_Network_I	no	no	yes	yes	no	no
Computer_Network_II	no	no	no	yes	no	no
Network_Security	no	no	no	yes	no	no
Database_Engineering	no	no	no	no	no	no
Oracle	no	no	no	no	no	no
Software_Enfineering	no	no	no	no	no	yes
Software_Testing_&_Quality_Assurance	no	no	no	yes	no	no
Computer_Graphics	no	no	no	yes	no	yes
Artficial_Intelligence	no	no	no	yes	no	no
Human_Computer_Interfaces	no	no	no	no	no	no
Hyper_Text_Markup_Language	no	no	no	yes	no	no
Javascript	no	no	no	no	no	no
Java_Server_Pages	no	no	no	no	no	no
Asp_Dot_Net	no	no	no	yes	no	no
Visual_Basic	no	no	yes	yes	yes	no
C_Programming	no	no	yes	yes	yes	no
Advanced_C	no	no	no	no	no	no
Object_Oriented_Design_&_Programming	no	no	no	yes	no	no
Java_Programming	yes	yes	yes	yes	no	no
Advanced_Java	no	no	no	yes	no	no
C_Sharp	no	no	no	yes	no	no
Data_Mining	no	no	no	no	no	no

=== Clusters for training data ===
Clustered Instances
0 333 (48%)
1 155 (22%)
2 32 (5%)
3 96 (14%)
4 81 (12%)

******The result of Cluster0 ******

Minimum support: 0.7 (233 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 6

Best rules found:
1. Visual_Basic=no 262 Discrete_Mathematical_Structure=no 241 conf:(0.92)
2. Discrete_Mathematical_Structure=no Design_&_Anaylisis_Of_Algorithm=no 260 Formal_System_And_Automata=no 238 conf:(0.92)
3. Javascript=no 268 Advanced_C=no 244 conf:(0.91)
4. Microprocessor=no 263 Discrete_Mathematical_Structure=no 239 conf:(0.91)
5. Design_&_Anaylisis_Of_Algorithm=no Formal_System_And_Automata=no 262 Discrete_Mathematical_Structure=no 238 conf:(0.91)
6. Switching_Theory_&_Logic_Design=no261 Discrete_Mathematical_Structure=no 237 conf:(0.91)
7. Formal_System_And_Automata=no 290 Discrete_Mathematical_Structure=no 263 conf:(0.91)
8. Discrete_Mathematical_Structure=no Formal_System_And_Automata=no 263 Design_&_Anaylisis_Of_Algorithm=no 238 conf:(0.9)
9. Data_Mining=no 273 Advanced_C=no 247 conf:(0.9)
10. Formal_System_And_Automata=no 290 Design_&_Anaylisis_Of_Algorithm=no 262conf:(0.9)

******The result of Cluster1 ******

Minimum support: 0.85 (131 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 3

Best rules found:
1. Javascript=no 142 Human_Computer_Interfaces=no 135 conf:(0.95)

(continued on next page)

2. Artficial_Intelligence=no 138 Human_Computer_Interfaces=no 131 conf:(0.95)
3. Data_Mining=no 143 Human_Computer_Interfaces=no 135 conf:(0.94)
4. Software_Testing_&_Quality_Assurance=no 142 Human_Computer_Interfaces=no 133 conf:(0.94)
5. Human_Computer_Interfaces=no 145 Javascript=no 135 conf:(0.93)
6. Human_Computer_Interfaces=no 145 Data_Mining=no 135 conf:(0.93)
7. Javascript=no 142 Data_Mining=no 132 conf:(0.93)
8. Data_Mining=no 143 Javascript=no 132 conf:(0.92)
9. Software_Testing_&_Quality_Assurance=no 142 Data_Mining=no 131 conf:(0.92)
10. Human_Computer_Interfaces=no 145 Software_Testing_&_Quality_Assurance=no 133 conf:(0.92)

******The result of Cluster2 ******

Minimum support: 0.85 (27 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 3

Best rules found:
1. Operating_System_I=yes 29 Java_Programming=yes 29 conf:(1)
2. C_Programming=yes 29 Java_Programming=yes 29 conf:(1)
3. Operating_System_I=yes C_Programming=yes 27 Java_Programming=yes 27 conf:(1)
4. Object_Oriented_Design_&_Programming=yes 28 Java_Programming=yes 27 conf:(0.96)
5. Java_Programming=yes 31 Operating_System_I=yes 29 conf:(0.94)
6. Java_Programming=yes 31 C_Programming=yes 29 conf:(0.94)
7. C_Programming=yes 29 Operating_System_I=yes 27 conf:(0.93)
8. Operating_System_I=yes 29 C_Programming=yes 27 conf:(0.93)
9. C_Programming=yes Java_Programming=yes 29 Operating_System_I=yes 27 conf:(0.93)
10. Operating_System_I=yes Java_Programming=yes 29 C_Programming=yes 27 conf:(0.93)

******The result of Cluster3 ******

Minimum support: 0.85 (82 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 3

Best rules found:
1. Asp_Dot_Net=no 86 Javascript=no 84 conf:(0.98)
2. C_Sharp=no 84 Javascript=no 82 conf:(0.98)
3. Data_Structure_I=yes 89 Javascript=no 86 conf:(0.97)
4. Formal_System_And_Automata=no 86 Javascript=no 83 conf:(0.97)
5. Software_Testing_&_Quality_Assurance=no 86 Javascript=no 83 conf:(0.97)
6. Computer_Network_II=no 85 Computer_Organisation=no 82 conf:(0.96)
7. Computer_Network_II=no 85 Javascript=no 82 conf:(0.96)
8. Computer_Organisation=no 88 Javascript=no 84 conf:(0.95)
9. Data_Mining=no 88 Javascript=no 84 conf:(0.95)
10. Human_Computer_Interfaces=no 87 Javascript=no 83 conf:(0.95)

******The result of Cluster4 ******

Minimum support: 0.85 (69 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 3

Best rules found:
1. Computer_Network_II=no 76 Computer_Graphics=yes 74 conf:(0.97)
2. System_Programming=no 72 Java_Server_Pages=no 70 conf:(0.97)
3. Computer_Network_II=no Java_Server_Pages=no 72 Computer_Graphics=yes 70 conf:(0.97)
4. Advanced_C=no 71 Java_Server_Pages=no 69 conf:(0.97)
5. Advanced_Java=no 71 Java_Server_Pages=no 69 conf:(0.97)
6. Computer_Graphics=yes 77 Computer_Network_II=no 74 conf:(0.96)
7. Formal_System_And_Automata=no 75 Java_Server_Pages=no 72 conf:(0.96)
8. Computer_Organisation=no 74 Computer_Graphics=yes 71 conf:(0.96)
9. Computer_Organisation=no 74 Java_Server_Pages=no 71 conf:(0.96)
10. C_Sharp=no 73 Computer_Network_II=no 70 conf:(0.96)

Out of 5 clusters, the result of cluster 2 is correct

Minimum support: 0.85 (69 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 3

Best rules found:
1. Operating_System_I=yes 29 Java_Programming=yes 29 conf:(1)
2. C_Programming=yes 29 Java_Programming=yes 29 conf:(1)
3. Operating_System_I=yes C_Programming=yes 27 Java_Programming=yes 27 conf:(1)
4. Object_Oriented_Design_&_Programming=yes 28 Java_Programming=yes 27 conf:(0.96)
5. Java_Programming=yes 31 Operating_System_I=yes 29 conf:(0.94)
6. Java_Programming=yes 31 C_Programming=yes 29 conf:(0.94)
7. C_Programming=yes 29 Operating_System_I=yes 27 conf:(0.93)
8. Operating_System_I=yes 29 C_Programming=yes 27 conf:(0.93)
9. C_Programming=yes Java_Programming=yes 29 Operating_System_I=yes 27 conf:(0.93)
10. Operating_System_I=yes Java_Programming=yes 29 C_Programming=yes 27 conf:(0.93)

4.6.2. Space complexity

Space complexity of Apriori association algorithm is $O(n + C_A)$, where n is the number of records and C_A is the constant for Apriori association rule algorithm.

Space complexity of Simple K-means algorithm is $O(n + C_S)$, where n is the number of records and C_S is the constant for Apriori association rule algorithm.

So space complexity of algorithm using combination of Simple K-means clustering and Apriori association rule algorithm is given in Fig. 7.

5. Experimental result

Table 7 shows the result using Weka as well as result of combined approach using Simple K-means clustering and Apriori association rule algorithm. A row numbered one presents the result before data preparation strategy. As all association rules are negative, we need to prepare the data explained in paper [16].

As this is Course Recommendation System, we need all positive association rules having "yes". So the result after data preparation is shown in row numbered two which consist of positive association rules having "yes". As we increase the support, we get refined rules but number of rules is less which is shown in row numbered three of Table 3.

Result of combined approach using Simple K-means clustering and Apriori association rule algorithm is shown in row numbered three of Table 7. We divide the data in five clusters and apply Apriori association rule algorithm on each cluster. Out of five clusters, the result of cluster 2 is correct as it contains all positive association rules having "yes". As in row numbered three of Table 7, for support value 0.85, we get 10 rules and as in row numbered three of Table 3, results using Apriori association rule algorithm, for support value of 0.6, we get only one rule.

6. Conclusion and future work

In this study we have compared the result of Apriori association rule algorithm which is existing algorithm in open source data mining tool Weka and other combination of clustering and association rule algorithm with the result of combined algorithm using Simple K-means clustering and Apriori association rule algorithm. Our approach uses the combination of clustering technique – Simple K-means and association rule algorithm – Apriori and finds the result. The result of combined algorithm are better than the result we obtain using Apriori association rule algorithm and other combination of clustering and association rule algorithm. The result using Apriori association rule algorithm needs the data preparation stage as well as if we increase the support then number of association rules are less. Also the results using combination of Expectation Maximization clustering and Apriori association rule algorithm and the combination of Farthest First clustering and Apriori association rule algorithm does not match with the real world choices of students. But if we use combined approach using Simple K-means clustering and Apriori association rule algorithm then there is no need to use data preparation stage and the number of association rules are more. This combined approach increases the strength of the association rule. So this Course Recommendation System would help the students to select proper course combination according to their interests.

This system can be also used to recommend courses to the student in distance learning Future work consist of finding other combination of data mining techniques which may be the combination of classification and association rule algorithm or combination of classification, clustering and association rule algorithm. This work can also be further extended and developed so that it can be integrated with existing E-Learning System. This Course Recommendation System could help in building intelligent recommender system. This approach of recommending courses to new students can be immensely be useful in "MOOC (Massively Open Online Courses)".

References

[1] Margaret H. Dunham, Data Mining Introductory and Advanced Topics, Pearson Education India, 2006. ISBN-8177587854, 9788177587852.
[2] F. Castro, A. Vellido, A. Nebot, F. Mugica, Applying data mining techniques to e-learning problems: a survey and state of the art, Evolution of Teaching and Learning Paradigms in Intelligent Environment. Studies in Computational Intelligence 62 (2007).
[3] C. Romero, S. Ventura, E. Garcia, Data mining in course management systems: MOODLE case study and tutorial, Computers and Education 51 (2007) 368–384.
[4] K. Seki, W. Tsukahara, T. Okamoto, System development and practice of e-learning in graduate school, in: Fifth IEEE International Conference on Advanced Learning Technologies, ICALT'05, IEEE Computer Society, Los Angeles, CA, 2005, pp. 740–744.
[5] O.R. Zaïane, J. Luo, Towards evaluating learners' behavior in a web-based distance learning environment, in: IEEE International Conference on Advanced Learning Technologies, ICALT'01, Madison, WI, August 6–8, 2001, pp. 357–360.
[6] C. Carmona, G. Castillo, E. Millán, Discovering student preferences in e-learning, in: Proceeding of the International Workshop on Applying Data Mining in e-Learning, 2007, pp. 33–42.
[7] W. Hamalainen, J. Suhonen, E. Sutinen, H. Toivonen, Data mining in personalizing distance education courses, in: World Conference on Open Learning and Distance Education, Hong Kong, 2004, pp. 1–11.
[8] Zamil Itmaz, Miguel Megias, Using recommendation system course management system to recommend learning objects, International Arab Journal of Information Technology 5 (3) (2008) 234–240.
[9] Sunita B. Aher, L.M.R.J. Lobo, A Framework for Recommendation of courses in E-learning System, International Journal of Computer Applications 35 (4) (2011) 21–28.
[10] Sunita B. Aher, L.M.R.J. Lobo, Data mining in educational system using WEKA, in: Proceedings on International Conference on Emerging Technology Trends (ICETT), IJCA (International Journal of Computer Application) (3) (2011) 20–25.
[11] http://www.cs.waikato.ac.nz/ml/weka/ (accessed 16.05.12).
[12] Sunita B. Aher, L.M.R.J. Lobo, A comparative study of association rule algorithms for course recommender system in e-learning, International Journal of Computer Applications 39 (1) (2012) 48–52.
[13] http://en.wikipedia.org/wiki/Expectation%E2%80%93maximization_algorithm (23.01.12).
[14] Xuezhi Zheng, Zhihua Cai, Qu Li, An experimental comparison of three kinds of clustering algorithms, in: Proceedings of International Conference on Neural Network and Brain, ICNN&B'05 in IEEE Xplore, 2005, pp. 767–771.
[15] Sunita B. Aher, L.M.R.J. Lobo, A comparative study for selecting the best unsupervised learning algorithm in e-learning system, International Journal of Computer Applications 41 (3) (2012) 27–34.
[16] Sunita B. Aher, L.M.R.J. Lobo, Data preparation strategy in e-learning system using association rule algorithm, International Journal of Computer Applications 41 (3) (2012) 35–40.
[17] Markus Hegland, The Apriori Algorithm – A Tutorial, WSPC/Lecture Notes Series, 2005.
[18] K. Verbert, N. Manouselis, X. Ochoa, M. Wolpers, Context-aware recommender systems for learning: a survey and future challenges, IEEE Transactions on Learning Technologies 5 (4) (2012).
[19] J. Lu, A personalized e-learning material recommender system, in: Proceedings of the 2nd International Conference on Information Technology for Application (ICITA 2004), Macquarie Scientific Publishing, 2004.
[20] Osmar R. Zaïane, Building a recommender agent for e-learning systems, in: Proceedings of the International Conference in Education, Auckland, New Zealand, 2002, pp. 55–59.
[21] Mohamed Koutheaïr Khribi, Mohamed Jemni, Olfa Nasraoui, Automatic recommendations for e-learning personalization based on web usage mining techniques and information retrieval, Educational Technology and Society 12 (4) (2009) 30–42.
[22] E. García, C. Romero, S. Ventura, C. de Castro, A collaborative educational association rule mining tool, The Internet and Higher Education 14 (2) (2011) 77–88. Web mining and higher education: Introduction to the special issue.
[23] T. Tang, G. McCalla, Smart recommendation for an evolving e-learning system: architecture and experiment, International Journal on E-Learning 4 (1) (2005) 105–129.
[24] S.W. Changchien, T. Lu, Mining association rules procedure to support on-line recommendation by customers and products fragmentation, Expert Systems with Applications 20 (2001) 325–335.
[25] K.W. Cheung, J.T. Kwok, M.H. Law, K.C. Tsui, Mining customer product ratings for personalized marketing, Decision Support Systems 35 (2003) 231–243.
[26] J.K. Kim, Y.H. Cho, W.J. Kim, J.R. Kim, J.H. Suh, A personalized recommendation procedure for Internet shopping support, Electronic Commerce Research and Applications 1 (2002) 301–311.

[27] T.H. Cho, J.K. Kin, S.H. Kim, A personalized recommender system based on web usage mining and decision tree induction, Expert Systems with Applications 23 (2002) 329–342.
[28] O.B. Kwon, I know what you need to buy: context – aware multimedia-based recommendation system, In Expert Systems with Applications (2002).
[29] L. Lee, T.C. Lu, Intelligent agent-based systems for personalized recommendations in Internet commerce, Expert Systems with Applications 22 (2002) 275–284.
[30] R.D. Lawrence, G.S. Almasi, V. Kotlyar, M.S. Viveros, S. Duri, Personalization of supermarket product recommendations, Data Mining and Knowledge Discovery 5 (2001) 11–32.
[31] J.B. Schafer, J.A. Konstan, J. Riedl, E-commerce recommendation applications, Data Mining and Knowledge Discovery 5 (2001) 115–153.
[32] O.R. Zaiane, Web usage mining for better web-based learning environment, in: Prof. of Conference on Advanced Technology for Education, Banff, AB, June 2001, pp. 60–64.