International Journal of
Intelligent Systems and
Applications in Engineering
IJISAE
ISSN:2147-6799 http://ijisae. atscience.org
Advanced Technology and Science
Original Research Paper

Grade prediction improved by regular and maximal association rules

Anca Loredana Udristoiu *[1], Stefan Udristoiu [1]

Accepted 15th August 2014 **DOI: 10.18201/ijisae.17210**

Abstract: In this paper we propose a method of predicting student scholar performance using the power of regular and maximal association rules. Due to the large number of generated rules, traditional data mining algorithms can become difficult and inappropriate to educational systems. Thus, we use some methods to overcome this problem, discovering rules useful in educational process. These methods are applied to the e-learning system Moodle, for "Database" course.

Keywords: *Education data mining, Regular association rule, Maximal association rule, Learning management system.*

1. Introduction

The e-learning systems represent educational contexts where impressive data collections about the interactions between students and teachers, assignments, chats, forums, quizzes are produced. Using this information provided by an e-learning system, we could find interesting relationships between student performance and its course activity and interactions among other participants.
Sometimes, this exponential increasing could create difficulties in offering useful information to e-learning system participants.
In this case, the techniques from data mining could be used to offer methods for understanding, processing and modelling data, resolving the limitations of the e-learning systems.
The association rules are one of the data mining technique successfully applied in e-learning. Its application becomes difficult when the number of rules is great or when some interesting rules are omitted.
So, in this this paper we proposed the generation of rules using the CBA algorithm [1], besides the generation of maximal association rules that offer interesting relationships [2].
As an example, consider a database of students with different marks. Suppose that there is one mark, say m1, that is very common, appearing for 50% of the students, and another much less common mark, say m2, appearing in only 10% of the students. Suppose further that m1 is described by descriptors SD1, SD2, SD3 and that m2 is described by either SD1 or SD2, but not both. If we search for regular associations, we may get the association between m1 and SD1, SD2, SD3, but we would miss the mark association linking SD1 alone and SD2 alone to m2. The reason is that the many instances of m1 with SD1, SD2, SD3 reduce the confidence of the rules "SD1 ⇒ m2" and "SD2 ⇒ m2". In order to obtain the association "SD1 ⇒ m2" we need to capture the notion that whenever SD1 appears alone then m2 also appears, with high confidence. Regular association rules fail to capture such associations, whereas maximal association rules [2] are designed to capture these cases.
The objective of this study is to classify students before the final evaluation, at any moment of time chosen by the teacher, into four mark classes: excellent, good, average, insufficient.
In this way, all the learning and evaluation process could be improved.
The remainder of this paper is structured as follows. Section 2 presents the background of applying association rules in educational system; section 3 presents the data pre-processing; section 4 presents the generation of regular and maximal association rules and details the process of predicting marks. Finally, section 5 discusses the experiments and summarizes the conclusions of this study.

2. Background

Predicting students' performance is one of the most important and useful applications of educational data mining and its goal is to score or mark from student course behaviour and activity [3].
Different techniques of data mining have been applied for predicting students' performance, as: neural networks, Bayesian networks, rule-based systems, and regression and correlation analysis.
Among these techniques, association rule mining (ARM) is one of the most popular. Its objective is to discover patterns in datasets.
An excellent review about the application fields of ARM in educational problems was made in [4].
In [3], association rule mining using genetic programming is used to provide feedback to instructors from multiple-choice quiz data.
In [5,6], ARM is used to make automatic recommendation system for web-based learning environments that takes into account profiles of on-line learners, their access history and the collective navigation patterns.
The study [7] focuses on the discovery of interesting contrast rules, which are sets of conjunctive rules describing interesting characteristics of different segments of a population.
A computer-assisted approach to diagnosing student learning problems in science courses and offer students advice was presented in [8], based on the concept effect relationship, a specification of the association rules technique.
Also, association rules were applied for student learning

[1] *University of Craiova, Department of Computers and Information Technology blvd. Decebal, n. 107,Craiova, Romania*
* *Corresponding Author: Email: anca_soimu@yahoo.com*
Note :This paper has been presented at the International Conference on Advanced Technology&Sciences (ICAT'14) held in Antalya (Turkey), August 12-15, 2014.

assessments in [9, 10, 11, 12], course adaptation to the students' behaviour in [13, 14, 15] and evaluation of educational Web sites [16].

3. Student data processing

We have collected data from on-line course activity provided by Moodle [20] that is one of the most widely used open source learning management system. In fact, we have used the following data based on student 'Database' course activity [3]:

- Nassignment – number of assignements taken.
- Nquiz - number of quizz taken.
- Nquiz_p - number of quiz passed.
- Nquiz_f - number of quiz failed.
- Nmessages - number of messages sent to the chat.
- Nmessages_ap - number of messages sent to the teacher.
- Nposts - number of messages sent to the forum.
- Nread - number or forum messages read.
- Total_time_assignment- total time spent on assignment.
- Total_time_quiz – total time used in quizzes.
- Total_time_forum- total time used in forum.
- Mark- final mark the student obtained in the course.

Since the data provided by Moodle are structured, they didn't necessitate preparation [3]. So, we directly discretise them, transforming numerical values into categorical ones for a good interpretation and understanding.

We have used the manual method for discretising all attributes, so the teacher has to specify the cut off points. The mark descriptor has four values:

- insufficient, if value < 5,
- average, if value > 5 and < 7,
- good if value >7 and < 9,
- excellent if value > 9.

The other attributes have the values: LOW, MEDIUM and HIGH [3].

A student is represent in Prolog by means of a term:

student(ListofDescriptors)

where the argument is a list of terms used to specify the student attributes. The term used to specify the student attributes is of the form:

descriptor(DescriptorName,DescriptorValue).

The model representation of students is in the following example:

student([
descriptor(Nassignment, medium),
descriptor(Nquiz, low),
descriptor(Nquiz_p, low),
descriptor(Nquiz_f, high),
descriptor(Nmessages, medium),
descriptor(Nmessages_ap, medium),
descriptor(Nposts,low),
descriptor(Nread, low),
descriptor(Total_time_assignment, low),
descriptor(Total_time_quiz, low),
descriptor(Total_time_forum, low)
]).

4. Associative classifiers

Association rule mining is one of the most important tasks in data mining and initially, methods applied to market basket analysis, were developed. So, the generation of association rules was introduced in [17] and the algorithm AIS was proposed. In [18], the algorithm called SETM was proposed to discover association rules using relational operation. In [19], the algorithms called Apriori and AprioriTid were proposed, bringing important improvements to older methods.

In this paper, the associative classifiers are used to define rules that classify students in four categories based on their activities.

The proposed method considers that the dataset contains N cases, described by 11 categorical parameters. These N cases have been classified into four categories/classes. Let $S=\{r_1, r_2,..., r_n\}$ be a set transactional students, called items. A transaction, *t*, over *S*, is a subset $t \subseteq S$. A *database*, *D*, over *S* is a multiset of transactions over *S*. A *grouping*, *G*, of *S* is a division of *S* into disjoint sets, $G=\{g_1, g_2, ..., g_{k,}\}$ *where* $k =1..4$. We call the elements of *G* *categories*. For an item *r*, *g(r)* represents the category that contains the *r* item [2].

An *association rule* is a rule of the form $X \Rightarrow Y$, where *X* and *Y* are disjoint sets of items from *S*. The support and confidence of an association rule is defined as in [17].

In a maximal association rule $X \Rightarrow_{\max} Y$ we are interested in capturing the notion that whenever *X* appears *alone* then *Y* also appears, with some confidence.

The rule has the confidence c, if c% of cases in D that contain X are labelled with y class. The rule has the support s, if s% of the cases in D contain X and are labelled with y class.

The main objective is to generate a set of ARs that satisfy the specified minimum support(*minsup*) and minimum confidence (*minconf*), and to classify students using the ARs.

Associative classifiers are a two-stage approach of classification, in which a set of association rules between the students descriptors and mark categories is first discovered and then a compact classifier is created by selecting the most important rules for classification.

4.1. Regular rule generation

The proposed method uses a modified version of CBA algorithm [1], for discovering the rules between students' descriptors and mark categories.

A rule is represented using a Prolog fact:

rule(Mark, Score, ListofStudentDescriptors).

where Score is the rule confidence, the body of the rule, is composed by conjunctions of student descriptors, while Mark, the head of the rule, is the mark category.

The student modelling in terms of itemsets and transactions is the following:

- the set of marked students of the training data represent the transaction set, D.
- the itemsets are formed by descriptors, so an item is represented by a pair (descriptor, value).
- the frequent itemsets represent the itemsets with the support greater than or equal to the minimum support defined (minsup).
- the itemsets of cardinality between 1 and k are iteratively found, where k represents the maximum length of an itemset; in our case, k is the number of descriptors, namely eleven.
- for a rule of the form SD ⇒ m, SDCount represents the number of cases in the transactional set, D that contain the descriptor set, SD.
- for a rule of the form SD ⇒ m, ruleCount represents the number of cases in the transactional set, D, which contain the descriptor set, SD and are labelled with the mark category m.
- the support is (ruleCount/|D|)*100%.
- the confidence is (ruleCount/SDCount)*100%.
- the frequent itemsets are used for rule generation.

The generation of frequent itemsets is illustrated by Algorithm 1.

Algorithm. 1: Frequent set generation on the training set of

students.

Input: the transactional set, D, containing the students with various mark classes; each student is described by a set of k descriptors, SD; the defined minimum support, minsup, and the defined minimum confidence, minconf.

Output: the set of frequent itemsets F_k.

Method:

```
F1 = {frequent 1-itemsets};
for(k=2;Fk-1≠ null; k++) do
 {Ck = Fk-1 join {SDk};
 for each distinct markClass  m ∈ D do
      {
       for each c ∈ Ck do
      {c.SDCount++;
            if c.markClass = m then
                  c.ruleCount++;
       }
      }
    }
Fk={c ∈ Ck|c.support ≥ minsup}
```

In the first step, the algorithm selects the frequent itemsets of 1-length. The maximum length of itemsets is for k=11. For each step (lines 2-12), the algorithm performs the following operations:

- the frequent itemsets F_{k-1} found in the (k-1) step are joined to the values of the descriptor, SD_k, to generate the itemsets C_k (line 3).

-it scans the transactional database and updates the support and confidence of itemsets from C_k (lines 4-11).

-new frequent itemsets F_k are found (line 12).

The generation of rules based on frequent itemsets F_k includes the following steps:

- for all the rules that have the same descriptor set (SD), the rule with the highest confidence is chosen.
- accurate rule: confidence   minconf.

As an example of a generated rule by applying the Algorithm 1, supposing that the minimum support is 20%, and the minimum confidence is 66.7%, is:

Rule
(good, 80,
[descriptor(Nassignment, high),
descriptor(Nquiz, high),
descriptor(Nquiz_p, medium)]).

Rule(average, 80,
[descriptor(Nassignment, medium),
descriptor(Nquiz, medium),
descriptor(Total_time_assignment, low),
descriptor(Total_time_quiz, medium
]).

4.2. Maximal association rules generation

For a transaction *t*, a category g_i and an itemset $X \subseteq g_i$, we say that *X is alone in t*, if $t \cap g_i = X$, meaning that X is alone in *t*, if *X* is the largest subset of g_i which is in *t* [2].

A maximal association rule, or M-association, is a rule of the form $X \Rightarrow_{\max} Y$, where *X* and *Y* are subsets of distinct categories, $g(X)$ and $g(Y)$, respectively. The M-support of the maximal association, denoted by s_{max} is defined as:

$$s_{\max} = \left|\{t : t \text{ supports maximally X and t supports Y}\}\right|$$

The M-confidence of the maximal association, denoted by c_{max} is defined as:

$$c_{\max} = \frac{s_{\max}(X \Rightarrow_{\max} Y)}{\left|D(X, g(Y))\right|}$$

,where *D(X, g(Y))* is the subset of the database *D* consisting of all the transactions that maximally support *X* and contain at least one element of *g(Y).*

We search for associations where the M-support is above some user-defined minimum support, and the M-confidence is above some user-defined minimum confidence. A set X with M-support at least equal to the minimum support is said to be M-frequent.

It can be observed that in the definition of maximal association rules the antecedent is maximal, but the consequent need not to be maximal, but alternative definitions are also possible [2].

Consider the following database *D* consisting of the 10 transactions, as in Table I:

Table 1. An example of a database with transactions

ID	*Transaction*
1	(Nassignment, high), (Total_time_assignment, medium), good, excellent
2	(Nassignment, high), (Total_time_assignment, medium), (Total_time_forum, low), insufficient, average
3	(Nassignment, high), (Total_time_assignment, medium), (Total_time_forum, low), average
4	(Nassignment, high), (Total_time_assignment, medium), (Total_time_forum, low), good, average
5	(Nassignment, high), (Total_time_assignment, medium), good
6	(Nassignment, high), (Total_time_assignment, medium), good
7	(Nassignment, high), (Total_time_assignment, medium), (Total_time_forum, low), average, insufficient
8	(Nassignment, high), (Total_time_assignment, medium), good
9	(Nassignment, high), (Total_time_assignment, medium), (Total_time_forum, low), average
10	(Nassignment, high), (Total_time_assignment, medium), (Total_time_forum, low), insufficient

We group the elements into two categories, G={descriptors, marks}.

By establishing the minimum support to 20% and minimum confidence to 66.7%, the following maximal association rules are obtained:

Rule
(good, 100,
[descriptor(Nassignment, high),
descriptor(Total_time_assignment, medium)]).

Rule
(average, 83,
[descriptor(Nassignment, high),
descriptor(Total_time_assignment, medium),
descriptor(Total_time_forum, low)]).

As regular rule, the first rule has the confidence 47%, thus with a 66.7% confidence threshold, it is not obtained as a regular association rule.

Computing maximal association rules is faster than computing regular associations, because for each mark, any transaction M-supports at most one itemset. The steps to reach the maximal frequent itemsets are outlined in the Algorithm 2.

Algorithm. 2: Finding all maximal frequent sets.
Input: the database D with transactions, the minimum support s_{min}.
Output: the sets Msets containing all maximal frequent sets.
Method:
$M_{sets} = \Phi$
for t $\in$ D do {
  for g $\in$ G do {
      X = t ∩ g
      if X ≠ Φ then
          Count(X)++
        if Count(X)> s_{min} then
            $M_{sets} = M_{sets} \cup \{X\}$
    }
  }

The steps to reach the maximal semantic rules are outlined in the Algorithm 3: for each M-frequent set *X* of student descriptors and mark *g*, generates the sub-databases *D′* of *D*(*X*, *g*) that consists of the projection on mark *g* of the transactions M-supporting *X*. The M-frequent sets constitute the body (left-hand-side) of the rules. After that, the algorithm computes the head of the rules (the right-hand-side). Consider an M-frequent set *X*, and suppose $X \Rightarrow_{\max} Y$ is an M-association, where $Y \subseteq g$. Only the transactions *within* *D′* can possibly support the rule $X \Rightarrow_{\max} Y$. Moreover, suppose that the support of *Y* within *D'* is $s_{D'}(Y)$. Then the M-support of $X \Rightarrow_{\max} Y$ is $s_{D'}(Y)$, and the M-confidence of the rule is $s_{D'}(Y)/|D'|$. Thus, in order to find all M-associations with minimum M-support, s_{min}, and minimum M-confidence, c_{min}, we search within *D*' for all sets *Y* with support max(s_{min}, $c_{min} \cdot |D'|$). Generating all such frequent sets in *D*', *Frequent-Sets(D*', *s)*, is performed by Algorithm 1.

Algorithm. 3: Finding all maximal frequent sets.
Input: the database D with transactions, the minimum support s_{min} and the minimum confidence, c_{min}.
Output: the set of maximal pattern semantic rules, MaximalRules, and M-support, M-confidence of each maximal rule.
Method:
M = Msets
for X $\in$ M do{
  for g $\in$ G do{
      D'= D(X, g)
      s = max(s_{min}, $c_{min} \cdot |D'|$)
      Fsets = Frequent-Sets(D', s)
      for Y $\in$ Fsets do{
            MaximalRules = MaximalRules ∪ $\{X =>_{max} Y\}$
            M-support =$s_{D'}$(Y)
            M-confidence = $s_{D'}$(Y)/|D'|
        }
    }
 }

4.3. Student classification

The set of generated rules, *Rules,* represents the classifier. The classifier is used to predict which mark the student could obtain. Being given a new student, the classification process searches in the rules' set for finding its most appropriate mark.
The algorithm is described in the following steps:
Algorithm. 4: Algorithm for determining a student mark.
Input: new student, *S*, the set of generated rules, *Rules*; each rule has the confidence *Ruleconf.*
Output: the list of marks attached to the student, *S*.
Method:
MaxConf = 0
MarkSet = null
foreach rule R in Rules do{
   if (match(R, S) = 1) then{
        if (maxConf <= Ruleconf) then {
            Add rule R to MarkSet
            MaxConf = Ruleconf
        }
   }
}
*Diagnose the student, S, with the marks from MarkSet.
The algorithm verifies if the student, S, is matched to any rule, R from the Rules set, and the rules with maximum confidence are selected.
The function match(R,S) returns 1, if all the descriptors, which appear in the body of the rule are included in the descriptors of the rule, otherwise it returns 0:

$$\text{Match(R,S)} = \begin{cases} 1, \text{if descriptors(R)} \subseteq \text{descriptors(S)} \\ 0, \text{otherwise.} \end{cases}$$

where descriptors(R) represents the set of descriptors of the rule, R, and descriptors(S) represents the set of descriptors of student, S.

4.4. Experiments

In the experiments realized through this study, two databases are used for the learning and testing process. The database used to learn the correlations between student behaviour and marks, contains information about 40 students.
For each mark class, the following metrics (accuracy-A, sensitivity-S, specificity-SP) are computed in the case in which we consider only the regular semantic rules (R) and the other case in which we consider the regular and maximal semantic rules (R+M):

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN} \quad (1)$$

$$\text{Sensitivity} = \frac{TP}{TP + FN} \quad (2)$$

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (3)$$

where, TP represents the number of true positives (students correctly evaluated with the searched mark), FP represents the number of false positives (students incorrectly evaluated with the searched mark), TN represents the number of true negatives (students correctly evaluated with a different mark), FN represents the number of false negatives (students incorrectly evaluated with a different mark).
The results of the presented method are very promising as can be observed in Table II and the improvements were brought enriching the classifier by the maximal semantic rules.

5. Conclusion

In this study, methods based on ARM are proposed and developed to assist the teacher by doing the pre-evaluation of students during a course study. For establishing correlations with the mark, we experimented and selected some descriptors of the student activity in the Moodle system for a “Database” course.

The results of experiments are very promising and show that the methods based on ARM are very useful for predicting the results of the student during a course activity.

The Prolog language used for representation of students' descriptors and rules makes a simple and flexible integration of our methods with other learning management systems.

Table 2. Experimental results

Mark	*Accuracy (%)*		*Sensitivity (%)*		*Specificity (%)*	
	R	R+M	R	R+M	R	R+M
Excellent	95.5	97.1	92.7	94.1	72	72.9
Good	96.7	97	92.2	92.2	71.5	71.9
Average	96.1	96.2	90.5	91.5	72	72.5
Insufficient	96.1	96.1	90.5	91.8	71.4	71.9

In future work, it would be interesting to repeat the analysis using more data from different types of courses and also to select other student descriptors. It would be also very useful to do experiments using more experts in order to analyse the obtained rules for discovering interesting relashionships

Acknowledgements

This work was partially supported by the grant number 16C/2014, awarded in the internal grant competition of the University of Craiova.

References

[1] B. Liu, W. Hsu, Y. Ma(1998). Integrating classification and association rule mining. Proc. 4th International Conference on Knowledge Discovery and Data Mining. Pages. 80–86.

[2] A. Amir, Y. Aumann, R. Feldman, M. Fresco(2005). Maximal Association Rules: A Tool for Mining Associations in Text. Journal of Intelligent Information Systems. Vol. 25(3). Pages. 333–345.

[3] C. Romero, A. Zafra, J.M. Luna, S. Ventura(2013). Association rule mining using genetic programming to provide feedback to instructors from multiple-choice quiz data. Expert Systems. Vol. 30(2). Pages. 162-172.

[4] C. Romero and S. Ventura(2010). Educational data mining: a review of the state-of-the-art. IEEE Transactions on Systems, Man and Cybernetics-Part C: Applications and Reviews. Vol. 40. Pages. 601–618.

[5] O.R. Zaïane(2002). Building a Recommender Agent for e-Learning Systems. Proc. The International Conference on Computers in Education. ICCE. Pages. 55–59.

[6] O.R Zaïane, J. Luo(2001). Towards Evaluating Learners' Behavior in a Web-based Distance Learning Environment. Proc. IEEE International Conference on Advanced Learning Technologies, ICALT'01. Pages. 357–360.

[7] B. Minaei-Bidgoli, P.N. Tan, W.F. Punch(2004). Mining Interesting Contrast Rules for a Web-based Educational System. Proc. The 2004 International Conference on Machine Learning and Applications. ICMLA'04.

[8] G.J. Hwang, , C.L. Hsiao, C.R. Tseng(2003). A Computer-Assisted Approach to Diagnosing Student Learning Problems in Science Courses. Journal of Information Science and Engineering. Vol. 19. Pages. 229–248.

[9] A. Kumar(2005). Rule-Based Adaptive Problem Generation in Programming Tutors and its Evaluation. Proc. The 12th International Conference on Artificial Intelligence in Education. Pages. 36–44.

[10] T. Matsui, T. Okamoto(2003). Knowledge Discovery from Learning History Data and its Effective Use for Learning Process Assessment Under the e-Learning Environment. Proc.Society for Information Technology and Teacher Education International Conference. Pages. 3141–3144.

[11] S.D. Resende, V.M.T. Pires(2001). An Ongoing Assessment Model for Distance Learning. Proc. International Conference Internet and Multimedia Systems and Applications. Acta Press. Pages. 17–21.

[12] S.D. Resende, V.M.T. Pires(2002). Using Data Warehouse and Data Mining Resources for Ongoing Assessment of Distance Learning. Proc. IEEE International Conference on Advanced Learning Technologies. ICALT'02.

[13] M.F. Costabile, A. De Angeli, T. Roselli, R. Lanzilotti, P. Plantamura(2003). Evaluating the Educational Impact of a Tutoring Hypermedia for Children. Information Technology in Childhood Education Annual. Pages. 289–308.

[14] H.H. Hsu, C.H. Chen, W.P. Tai(2003). Towards Error-Free and Personalized Web-Based Courses. Proc. The 17th International Conference on Advanced Information Networking and Applications. AINA'03. Pages. 99–104.

[15] P. Markellou, I. Mousourouli, S. Spiros, A. Tsakalidis(2005). Using Semantic Web Mining Technologies for Personalized e-Learning Experiences. Proc. Conference on Web-based Education. WBE'05.

[16] M.L. Dos Santos, K. Becker(2003). Distance Education: a Web Usage Mining Case Study for the Evaluation of Learning Sites. Proc. The 3rd IEEE International Conference on Advanced Learning Technologies. ICALT'03. Pages. 360–361.

[17] R. Agrawal, T. Imielinski, and A. Swami(1993). Mining Association Rules between Sets of Items in Large Databases. Proc. SIGMOD International Conference on Management of Data. Pages. 207–216.

[18] M. Houtsma and A. Swami(1993). Set-Oriented Mining of Association Rules. Technical Report RJ 9567. IBM.

[19] R. Agrawal and R. Srikant(1994). Fast Algorithms for Mining Association Rules in Large Databases. Proc. International Conference on Very Large Data Bases. Pages. 487-499.

[20] J. Cole(2005). Using Moodle. O'Reilly.