Extraction Rules from Educational Data using FP Growth Algorithm with Correlation

Aye Myint Myat, Zaw Tun
University of Computer Studies, Yangon
ayemyintmyat2050@gmail.com; zawtun78@gmail.com

Abstract

Educational data mining is discovering knowledge from data that come from educational environment. This paper presents finding interesting patterns from educational database. Learning how student behaviors relate to academic results, could improve the teaching system. Association rule mining is used to find the interesting patterns, from which student behaviors can be learned. Correlation value, measured by lift ratio is used to measure the interestingness. The statistical index of the degree to which two variables are associated is the correlation coefficient. The lift value of greater than 1 indicates a positive correlation between antecedent and consequent. FP Growth algorithm is used to implement the association rule.

1. Introduction

Data mining is an important data analysis methodology that has been successfully employed in many domains. Association Rules represent discovering interesting patterns from the transaction database. It is well known as Market Basket analysis. Support and Confidence are two important measures in association rules. Another important concept in association rules is that of the Lift of the rule. With lift ratio, rules with strong correlation can be extracted.

Education Data Mining is the process of discovering knowledge from data come from educational environments. Data are collected from historical and operational data, which reside in the databases of educational institutes; (personal or academic). Also it can be collected from e-learning systems. The discovered knowledge can be used to better understand students' behavior, to improve teaching and many other benefits. The proposed system presents education data mining by Association rule mining algorithm. The discovered knowledge can be used to better understand students' behavior, to assist instructors, to improve teaching, to evaluate and improve e-learning systems, to improve curriculums and many other benefits.

This paper is organized as follows. Section 1 presents the introduction of the paper, in section 2 related work is described. Association rule mining is presented in section 3. In section 4, FP-Growth algorithm is described. Section 5 illustrates the proposed system design and in section 6, system implementation and sample case study for generating association rules from student database by FP-Growth algorithm are presented. Section 7 summarizes about the proposed system with conclusion.

2. Related Work

There are many works in educational data mining. (Romero, Ventura) presented a survey on educational data mining between 1995 and 2005[8]. They concluded that educational data mining is a promising area of research and it has a specific requirements not presented in other domains.

(Merceron, Ventura) gave a case study that used educational data mining to identify behavior of failing students to warn students at risk before final exam [6]. (Romero, Ventura, Garcia) gave another case study of using educational data mining in Moodle course management system [9]. They used each step in data mining process for mining e-learning data. Also, educational data mining used by (Minaei-Bidgoli, Kashy, Kortemeyer) to predict students' final grade using data collected from Webbased system [7]. (Beikzadeh, Delavari) used educational data mining to identify and then enhance educational process in higher educational system which can improve their decision-making process [3]. (Waiyamai) used data mining to assist development of new curricula, and to help engineering students to select an appropriate major [10].

Frequent-pattern mining plays an essential role in mining associations. Most of the previous studies, adopt an Apriori-like approach (Agarwal)[1]. The Apriori heuristic achieves good performance gained by reducing the size of candidate sets. However, in situations with a large number of frequent patterns, long patterns, or quite low minimum support thresholds, an Apriori-like algorithm may suffer from candidate generation and multiple database scan.

3. Association Rules

Association rules are used to identify relationships among a set of items in database. These relationships are based on co-occurrence of the data items. [2]

Mining association rules are to find interesting association or correlation relationships among a large set of data, i.e., to identify sets of attributes values (predicate or item) that frequently occur together, and then formulate rules that characterize these relationships. Mining association rules is composed of the following two steps –

- Discover the large itemsets, i.e., all sets of itemsets that have transaction support above a predetermined minimum support s.
- Use the large itemsets to generate the association rules for the database.

The overall performance of mining association rules is in fact determined by the first step. Let I= {i_l, i_2, . . . i_m } be a set of literals, called items. Let D be a set of transactions, where each transaction, T is a set of items such that $T \subseteq I$. Each transaction is associated with an identifier, called TID. Let X be a set of items. A transaction T is said to contain X if and only if $X \subseteq T$. An association rule is an implication of the form $X \Rightarrow Y$, where $X \subset I$, $Y \subset I$ and $X \cap Y = \phi$. The rule $X \Rightarrow Y$ holds in the transaction set D with confidence c if c% of transactions in D that contain X also contain Y. The rule $X \Rightarrow Y$ has support s in the transaction set D if s % of transactions in D contains $X \cup Y$.

3.1 Correlation

Lift ratio is used to represent the correlation of association rules. The lift value is the ratio of the confidence of the rule and the expected confidence of the rule. The lift is measured as the ratio of the probability of antecedent and consequent occurring together to the probability of antecedent and consequent occurring independently. The lift value of greater than 1 indicates a positive correlation between antecedent and consequent. With the lift value, we can measure the importance of a rule. The first rule, with the highest lift which means highest correlation is the most important, and so on.

Lift $(A \rightarrow B)$ = Support $(A \cup B)$ / Support (A) * Support (B)

4. FP-Growth

Candidate-generation-and-test operation is the bottleneck for Apriori-like methods. FP-Growth algorithm constructs FP Tree to store complete but no redundant information for frequent pattern mining and frequent patterns are mined from FP-Tree to get the frequent patterns.

FP-Tree consists of

- one root, labeled as “root”
- A set of items prefix, sub-trees as the children of the root
- Frequent-item header table

Item prefix sub-tree consists of three fields:

- item-name, registers which item this node represents
- count, registers the number of transactions represented by the portion of the path reaching this node
- node-link, links to the next node in the FP-tree carrying the same item-name, or null if there is none.

Frequent-item header table consists of two fields,

- item-name and
- head of node-link,

Sample Transaction database is shown in Table 1.

Table 1: Example Transaction

TID	Items Bought	(Ordered) Frequent Items
100	f, a, c, d, g, i, m, p	f, c, a, m, p
200	a, b, c, f, l, m, o	f, c, a, b, m
300	b, f, h, j, o	f, b
400	b, c, k, s, p	c, b, p
500	a, f, c, e, l, p, m, n	f, c, a, m, p

For the above transaction, FP Tree will be constructed as in Figure 1.

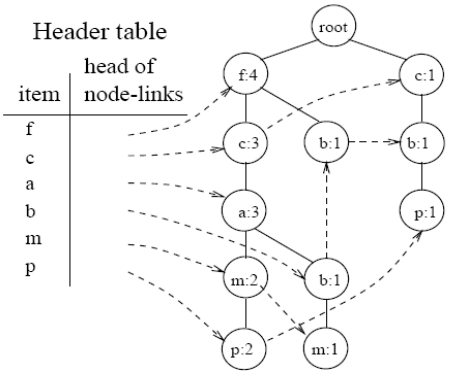

Figure 1: FP Tree Construction of Table 1

4.1. Implementing FP-Growth Algorithm

FP-Growth algorithm mines frequent patterns from a set of transactions in TID-itemset format {TID: itemset}, where TID is a transaction-id and itemset is the set of items bought in transaction. FP-Growth algorithm is implemented as follows:

Algorithm: FP-Growth. Mine frequent itemsets using an FP-tree by pattern fragment growth
Input: D, a transaction database;
min_sup, the minimum support count threshold.
Output: Complete set of frequent patterns
Method:
1. FP-tree is constructed in the following steps:

- Scan the transaction database D once. Collect F, the set of frequent items, and their support counts. Sort F in support count descending order as L, the list of frequent items.
- Create the root of an FP-tree, and label it as "null". For each transaction Trans in D do the following. Select and sort the frequent items in Trans according to the order of L. Let the sorted frequent item list in Trans be [p|P], where p is the first element and P is the remaining list. Call insert_tree([p|P], T), which is performed as follows. If T has a child N such that N.item-name = p.item-name, then increment N's count by 1; else create a new node N, and let its count be 1, its parent link be linked to T, and its node-link to the nodes with the same item-name via the node-link structure. If P is nonempty, call insert_tree(P, N) recursively.

2. The FP-tree is mined by calling FP-growth (FP-tree, null), which is implemented as follows.

```
Procedure FP-Growth (Tree, $\alpha$)
    If Tree contains a single path P then
    For each combination $\beta$ of the nodes in path P
    Generate pattern $\beta \cup \alpha$ with sup-count = min sup-count of nodes in $\beta$;
    Else for each $\alpha$i in the header of Tree {
    Generate pattern $\beta = \alpha_i \cup \alpha$ with sup-count = $\alpha$i.sup-count;
    Construct $\beta$'s conditional pattern base and then $\beta$'s conditional FP-tree Tree$_\beta$;
    If Tree$_\beta \neq \emptyset$ then
    Call FP-Growth (Tree$_\beta$, $\beta$);
```

5. Proposed System

Extracting interesting patterns from education database is presented in the proposed system. Education database includes academic and student profiles (personal records); for example Attendance, exercise, homework, resource-usage, grades, etc. Association mining algorithm is used to extract interesting relationships among attributes from the given dataset. It allows finding rules of the form If antecedent then (likely) consequent where antecedent and consequent are itemsets. Because, we are looking for items that characterize the final grade of students, consequent has one item which is final_grade= z where z is one value of the final grade such as distinction, credit, pass, fail, etc. Figure 2 presents the proposed system overview. It collects data from academic records, attendances and student profiles as the transaction database. Then FP-Growth algorithm is applied to transaction database and interest patterns are generated.

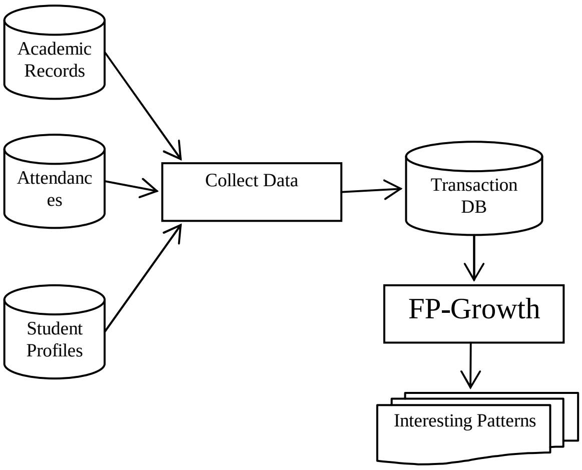

Figure 2: System overview

5.1 Process Flow of the System

The proposed system has following processes –

- It will collect data from different databases, student profiles (personal records), attendances and academic records into transaction database.
- Transaction database is applied into FP-Growth algorithm.
- According to FP-Growth algorithm, items are sorted by their support count.
- Then FP-Tree and header table is built.
- After mining the FP-Tree, we got the association rules an interestingness is measured by lift ratio.
- Then we got the relevant association rules, which have strong correlations

Process flow of the system is shown in Figure 3.

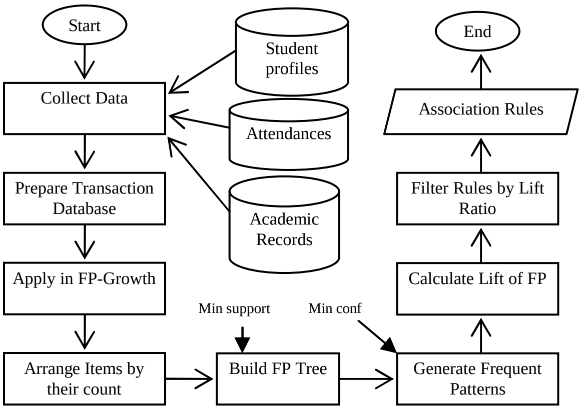

Figure 3: Process Flow of the System

6. System Implementation

The proposed system is implemented using Microsoft Visual Studio 2008. ASP .Net C# is used to implement the system. Student behaviors and their academic records are converted into transactional records in order to apply in the Association rule. Data attributes and their values are described in Table 2.

Table 2: Attributes and their values

Attribute	**Values**
Attendance	Good (100%-85%), Average(84%-65%), Fail(< 65%)
Tutorial	Excellent(100%-80%),Good(79%-65%),Pass(64%-40%),Fail(< 40%)
Resources-Usage	Yes(100%-50%), No(<50%)
Exercises-Done	Yes(100%-50%), No(<50%)
Homework-Done	Good(100%-75%), Average (74%-40%), Fail(< 40%)
Lab	Good(100%-75%),Average (74%-40%), Fail(< 40%)
Midterm	Excellent(100-80), Good(79-65), Pass(64-50), Fail(<50)
Final	Excellent(100-80), Good(79-65), Pass(64-50), Fail(<50)

6.1. Case Study

The sample case study used in this system is the small database containing 10 transactions. It is shown in Table 3.

Table 3: Small Database containing 10 transactions

No	**Attendance**	**Tutorial**	**Resource_Usage**	**Exercises**	**Homework**	**Mid Term**	**Lab**	**Final**
1	Good	Good	Yes	Yes	Good	Good	Good	Good
2	Good	Good	Yes	Yes	Good	Good	Good	Good
3	Good	Excellent	Yes	Yes	Good	Excellent	Good	Excellent
4	Average	Pass	No	Yes	Average	Pass	Average	Pass
5	Fail	Fail	Yes	No	Average	Pass	Average	Pass
6	Fail	Pass	No	No	Fail	Pass	Fail	Fail
7	Fail	Fail	No	No	Fail	Pass	Fail	Fail
8	Fail	Pass	No	No	Fail	Fail	Fail	Fail
9	Average	Pass	No	No	Average	Fail	Average	Pass
10	Good	Good	Yes	Yes	Good	Good	Good	Good

After applying into the FP-Growth algorithm, it has following rules as in Table 4.

Table 4: Example Association Rules

Patterns	**Support**	**Confidence**	**Lift**
Attendance = good, Resources_Usage = Yes, Exercise = Yes, Homework = Good, Lab=Good THEN Final = Good	0.3	100%	3.33
Attendance = Fail, Resources_Usage = No, Excercise = No, Homework = Fail, Lab = Fail THEN Final = Fail	0.2	100%	3.33

Lift value computation for above rules is shown below.

Attendance = good, Resources_Usage = Yes, Exercise = Yes, Homework = Good, Lab=Good → Final_Grade = Good) = 3 / 10 = 0.3
Attendance = good, Resources_Usage = Yes, Exercise = Yes, Homework = Good, Lab=Good = 3 / 10 = 0.3
Support (Grade = Good) = 3 / 10 = 0. 3
Confidence = Sup (X U Y) / Sup (X) = 0.3 / 0.3 = 1
Lift = Sup (X U Y) / Sup (X) * Sup (Y)
= 0.3 / (0.3 * 0.3) = 3.33

6.2. Experimental Results

In this system, there are 1700 transactions used to test the system. It finds out rules for both Midterm and Final exams. There are 140 rules for Midterm and 138 rules for Final when minimum support is set to 0.05. It is tested with different minimum support and system accuracy and rules generated are shown in Table 5.

Table 5: Results of the Systems

No.	Min-sup	Mid-Term rules	Final Rules	Accuracy
1	0.05	140	138	85.23%
2	0.15	104	89	87.18%
3	0.2	27	26	88.98%

7. Conclusion

The proposed system presents discovering interesting relationships in educational data. It learns how academic records are related to student behaviors in the pattern based manner. FP-Growth algorithm is used to find the interesting rules from education data. Interestingness is measured by association rule algorithm's two important measures support and confidence as well as lift ratio. It shows how useful data mining can be in higher education in particularly to improve student performance. Association Rules are sorted using lift metric so that we got more relevant rules.

8. References

[1] Agarwal, R., Aggarwal, C. and Prasad, V. V. V., "Depth-Firrst generation of large itemsets for association rules". IBM Tech. Report RC21538, July 1999.

[2] Agrawal , R. and Srikant, R., "Fast algorithms for mining association rules. In VLDB'94, pp. 487-499.

[3] Beikzadeh,M. and Delavari, N., "A New Analysis Model for Data Mining Processes in Higher Educational Systems". On the proceedings of the 6th Information Technology Based Higher Education and Training 7-9 July 2005.

[4] Han, J., “Data Mining: Concepts and Techniques”, Second Edition, ISBN 1-55860-489-8.

[5] Han, J., Pei J. Yin. Y and Mao, R. "Mining Frequent Patterns without candidate generation: A Frequent-Pattern Tree Approach", Data Mining and Knowledge Discovery, 8, 53–87, 2004.

[6] Merceron, A. and Yacef, K.,"Educational Data Mining: a Case Study" In Proceedings of the 12th International Conference on Artificial Intelligence in Education AIED 2005, Amsterdam, The Netherlands, IOS Press. 2005.

[7] Minaei-Bidgoli B., Kashy, D. Kortemeyer G., Punch W., "Predicting Student Performance: An Application of Data Mining Methods with an Educational Web-Based System". In the Processing of 33rd ASEE/IEEE conference of Frontiers in Education. 2003.

[8] Romero,C. and Ventura, S. ,"Educational data Mining: A Survey from 1995 to 2005".Expert Systems with Applications (33) 135-146. 2007.

[9] Romero, C. , Ventura, S. and Garcia, E., "Data mining in course management systems: Moodle case study and tutorial". Computers & Education, Vol. 51, No. 1. pp. 368-384. 2008.

[10] Waiyamai,K. "Improving Quality of Gradate Students by Data Mining" Department of Computer Engineering. Faculty of Engineering. Kasetsart University , Bangkok, Thailand. 2003.