Knowledge-Based Systems 24 (2011) 697–708

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Mining fuzzy specific rare itemsets for education data

Cheng-Hsiung Weng

Department of Management Information Systems, Central Taiwan University of Science and Technology, Taichung 406, Taiwan, ROC

ARTICLE INFO

Article history:
Received 29 July 2010
Received in revised form 17 December 2010
Accepted 13 February 2011
Available online 16 February 2011

Keywords:
Data mining
Association rules
Rare itemsets
Fuzzy sets
Learning problems

ABSTRACT

Association rule mining is an important data analysis method for the discovery of associations within data. There have been many studies focused on finding fuzzy association rules from transaction databases. Unfortunately, in the real world, one may have available relatively infrequent data, as well as frequent data. From infrequent data, we can find a set of rare itemsets that will be useful for teachers to find out which students need extra help in learning. While the previous association rules discovery techniques are able to discover some rules based on frequency, this is insufficient to determine the importance of a rule composed of frequency-based data items. To remedy this problem, we develop a new algorithm based on the Apriori approach to mine fuzzy specific rare itemsets from quantitative data. Finally, fuzzy association rules can be generated from these fuzzy specific rare itemsets. The patterns are useful to discover learning problems. Experimental results show that the proposed approach is able to discover interesting and valuable patterns from the survey data.

1. Introduction

Association rule mining is an important data mining approach that has been used to discover consumer purchasing behaviors from transaction databases [5,10]. Agrawal et al. [2] first discussed the problem of finding all rules from transaction data that satisfy the minimum support and the minimum confidence constraints. An association rule mining algorithm has two steps: (1) generate all frequent itemsets that satisfy the minimum support, and (2) generate all association rules that satisfy the minimum confidence from the already discovered frequent itemsets.

However, relatively infrequent data as well as frequent data exist in the real world. While the previous association rule discovery techniques can be used to discover some rules based on frequency, they are insufficient to determine the importance of a rule composed of data items based only on their frequency. If some data occur infrequently and these data appear simultaneously with other specific data in high proportions, the support is low even though the level of confidence is high, and thus this special characteristic is worth discovering. There are many interesting measures, such as *lift*, *all-conf*, *cosine* that can be used to discover meaningful association rules [10,29]. The existing association rules mining techniques are all designed for discovering rules in which both the support and confidence are high, rather than those where the support is low but the confidence is high. In this study, fuzzy specific-rare itemsets contain useful information. Fuzzy specific-rare itemsets are sets of items that rarely occur in the database together. An example is described below where a set of learning problems, such as a student's low quiz scores and homework scores, set them apart from the average student. Discovering such fuzzy specific-rare itemsets can be useful to teachers to find out which students need extra assistance.

Previous studies have tended to focus on the problem of discovering frequent patterns, meaning that only patterns that appear frequently in transaction data are mined. Patterns appearing in only a few data sets are not captured. In some cases, such as for the detection of computer virus attacks, fraudulent transactions in financial institutions or learning problems, those infrequent patterns (also known as rare patterns), are more interesting than frequent patterns [1].

It can be seen in Fig. 1 that all of the itemsets related to students' scores can be divided into 4 sections, A, B, C and D. Most of the existing algorithms focus on discovering frequent itemsets located in section C. Few of the existing algorithms focus on rare itemsets in sections $(A + B)$ and D. Since rare itemsets appear on either side of the itemset distribution, an extra measure is needed to distinguish them. A new threshold named *rank* is used to obtain specific rare itemsets. Fig. 1 shows that itemsets which are located in sections $(A + B)$ and D are rare itemsets. If we want to find specific rare itemsets, which are located in section A, we need to use the new *rank* threshold to obtain the specific rare itemset we want. The specific rare itemset is a kind of rare itemset but with lower ranks. To the best of our knowledge, there has been no study to discover only specific-rare itemsets as in section A.

Rare itemsets are given by all itemsets that are not extracted by the standard frequent itemset generation algorithms, such as Apriori or FP growth. Unlike the support measure for mining frequent itemsets, there is currently no measure possessing the

E-mail address: chweng@mgt.ncu.edu.tw

0950-7051/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.knosys.2011.02.010

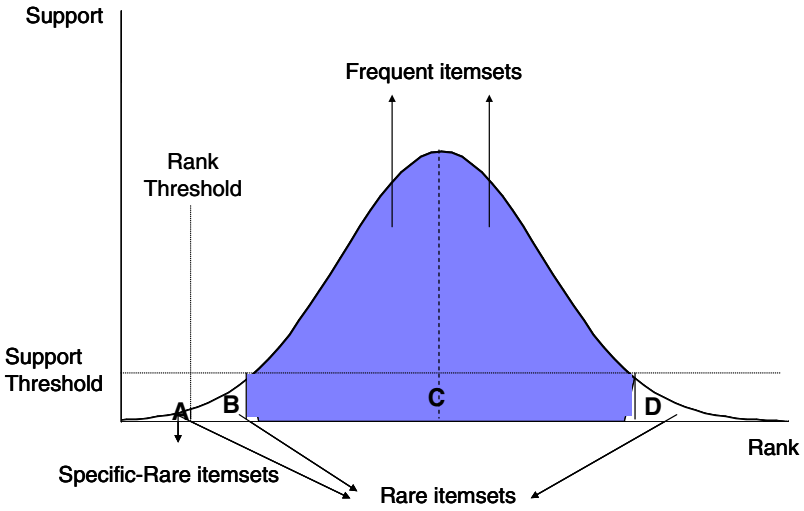

Fig. 1. Itemset distribution.

anti-monotone property that can be exploited for pruning the search space when generating rare itemsets [29]. Since there is no efficient solution for discovering rare itemsets, the existing algorithms and techniques are revised to tackle the problem of *specific-rare pattern* mining, especially for the purpose of detecting learning problems. In educational data, many attributes are recorded as quantitative data. Therefore, we propose an Apriori-based mining approach, Fuzzy Apriori Rare Itemset Mining (*FARIM*), for mining fuzzy specific rare itemsets consisting of quantitative data. Fuzzy association rules are generated from these fuzzy specific rare itemsets and the patterns used to detect learning problems.

The rest of this paper is organized as follows. Related work is reviewed in Section 2. The problem definitions are given in Section 3. The proposed algorithm is discussed in Section 4. In Section 5 the proposed method is demonstrated using an example. In Section 6 a case study based on survey data is carried out to demonstrate the usefulness of the proposed algorithm. Conclusions and future work are discussed in Section 7.

2. Related work

The previous approaches used for discovering itemsets can be divided into two types, namely frequent itemset mining and rare itemset mining techniques [1]. Whereas frequent itemset mining problems have been widely studied and many algorithms have been proposed, the problem of discovering rare itemsets has only just begun to draw interest. To the best of our knowledge, only a few algorithms have been developed to mine rare itemsets. In this section, the frequent and rare itemset approaches are briefly reviewed. In the following section, mining techniques used for educational data are reviewed and the difference between frequent itemset mining and rare itemset mining discussed.

2.1. Frequent itemset mining

The Apriori approach has been widely and successfully used to generate all frequent itemsets contained in a transaction database. Due to its great success and widespread usage, many variants of association rule mining algorithms have been proposed [7,8,11,18,19]. These algorithms can be roughly classified into three categories, according to the type of data they can handle: nominal/Boolean data [2,3], ordinal data [7], and quantitative data [8,9].

Since quantitative data are common in practical databases, a natural extension is to find association rules from quantitative data. To solve this problem, the value of a quantitative attribute can be partitioned into a set of intervals. The traditional algorithms for nominal data can then be applied. There are two major approaches to partitioning methods, crisp partitions and fuzzy partitions. Srikant et al. [27] were the first to propose a systematic method for partitioning quantitative data into intervals using crisp partitions. Based on the measure of *k-partial completeness*, they ensured that intervals were neither too big nor too small with respect to the set of rules they could generate. Srikant et al. [27] noted that their measure of interval quality does not work well on skewed data, because it may separate closed values that have the same properties. To address this problem, Miller and Yang [23] presented a new algorithm for finding rules that could adjust the distance-based quality and rule interest measures.

In fuzzy partitioning methods, the division of features into various linguistic values when mining association rules has been widely used in a variety of applications. For example, by Hong et al. [11–13] to find fuzzy association rules from quantitative transactions, and Hu et al. [15,16] to find fuzzy association rules from both quantitative and categorical attributes by dividing them into various linguistic values.

2.2. Rare itemset mining

The Apriori algorithm was developed to efficiently discover frequent itemsets [2]. Since then, new several algorithms such as the FP-Growth algorithm have been proposed for this purpose [10]. However, until now, few algorithms or approaches have been developed to discover rare patterns.

Szathmary et al. [28] proposed a method of discovering rare itemsets based on the Apriori algorithm which can be used to discover frequent itemsets. Briefly, the method can be divided into two steps: (1) all frequent itemsets and minimal rare itemsets are generated (i.e. *mRI*) through the apriori-like algorithm *MRG-Exp*, (2) the *mRI*s discovered in the first algorithm are taken as seeds for the input data for the second rare itemset mining algorithm *Arima*.

Adda et al. [1] proposed a framework for representing the different categories of interesting patterns and then instantiated it to the specific case of rare patterns. This approach, also based on the Apriori algorithm, uses a top-down strategy to discover patterns. The main idea is that if the itemset lattice representing the

itemset space in the classical Apriori approaches is traversed in a bottom-up manner, equivalent properties to the Apriori exploration of frequent itemsets are provided from which we can mine rare itemsets. This is a level wise exploration of the itemset space and includes an anti-monotone property, that is, if an itemset is not a rare itemset, its super-itemsets also cannot be rare itemsets.

Rare patterns are useful in various fields, such as medicine and biology. Masuda and Sakamoto [21] developed a framework to discover rare patterns which are difficult to discover from clinical data for dynamic evidence based medicine (DEBM). In the educational field normal behavior is very frequent, whereas abnormal or negative behavior is less frequent. Given a database where the behaviors of students in learning environments, such as schools, are recorded, it is likely that we will find that normal behaviors are represented by frequent patterns. Part of normal behaviors and all negative behaviors which appear infrequently are represented by rare patterns. In actuality though, it is the students exhibiting negative behaviors that need more assistance. Discovering these rare patterns related to negative student behaviors is helpful to teachers to identify which students might be having learning problems.

2.3. Association rule mining of educational data

Educational data mining is an emerging discipline, where methods are being developed to explore the unique types of data obtainable in the educational context. Romero and Ventura [25] applied data mining to traditional educational system data, particularly web-based courses, well-known learning content management systems, and adaptive and intelligent web-based educational systems. Romero et al. [26] described the process of mining e-learning data step-by-step, as well as how to apply the main data mining techniques, such as statistics, visualization, classification, clustering and association rule mining of Moodle data.

Association rule mining is one of the most well known mining methods. If–then statements concerning attribute values are produced based on the association of one or more attributes of a dataset with one or more other attributes. Association rule mining has been applied to web-based education systems. For example, Markellou et al. [20] proposed an ontology-based framework combining Apriori algorithms to determine which learning materials would be the most suitable for to the user. Minaei-Bidgoli et al. [24] discovered interesting contrast rules to identify attributes characterizing patterns of performance disparity between various groups of students. Yu et al. [31] applied fuzzy association rule mining techniques to find the relationships between learning behavior patterns, including the time spent online, number of articles read and published, number of questions asked, etc. Merceron and Yacef [22] used association rules and traditional SQL queries to discover common student mistakes to gain further insight into their learning performance. Association rule mining has been also applied to identify student learning problems. For instance, Hwang et al. [17] proposed a computer-assisted approach for diagnosing learning problems encountered by students in science courses and offering advice accordingly. Chen et al. [6] took the discovered association rules for common learning misconceptions and applied them to tune the structure of courseware. They modified the difficulty parameters in the courseware database so that learning pathways could be appropriately tuned. Hsu [14] developed an online personalized English learning recommendation system capable of providing ESL students with reading lessons suited to their own interests that would therefore increase their motivation to learn. Tseng et al. [30] combined fuzzy set theory, education theory and a data mining approach to find grade fuzzy association rules. They proposed a two-phase concept map construction (TP-CMC) algorithm with which to automatically construct a concept map of a course based upon historical testing records. However, in all of these studies, the focus was on mining frequent itemsets, rather than rare itemsets.

From the above, it can be seen that association rule mining helps to find those patterns which appear frequently in educational systems. In this study, however, the aim is to mine fuzzy rare itemsets from educational data so as to detect abnormal learning patterns. With the help of patterns composed of fuzzy rare itemsets, teachers can find out which students are having learning problems. After ascertaining this information, appropriate measures can be taken to assist these students.

3. Problem definition

In this section, we will define the problem of mining fuzzy rare itemsets from educational data. First, the items used in the proposed algorithm are introduced. Then, the membership degree for each different type of item is defined and used to calculate the support of itemsets from educational data. In a previous study [8], we defined how to calculate the support of the number itemsets. The student's score is a kind of number itemset. Here we use the same definitions as in our previous study for calculating the support of the number itemsets.

Definition 1. Let $IT = \{it_1, it_2, \ldots, it_m\}$ be a set of all items. An *s*-item is denoted as (it_i, s_i), where $it_i \in IT$ is the item name and s_i is the value of it_i. Semantically, (it_i, s_i) means that s_i represents the student's score for the ith item in the evaluation list.

Example 1. Assume that we have a data set containing 5 recorded scores, as shown in Table 1. The values "mid-term score", "final-term score", "quiz score", "homework score", and "attendance score" are abbreviated as *MS*, *FS*, *QS*, *HS*, and *AS*, respectively. In this example, there are five *s*-items named *FS*, *MS*, *QS*, *HS*, and *AS*. All the *s*-items are quantitative data, such as 91. An *s*-item, (*MS*, 91), means that the mid-term score of that student is 91.

Definition 2. An *r*-item can be a *linguistic s*-item. For simplicity, we use $b_i = (ic_i, f_i)$ to denote an *r*-item. An *r*-itemset *B* is a set of *r*-items, where all *r*-items must have distinct item names. We use $B = \{(ic_1, f_1), (ic_2, f_2), \ldots, (ic_n, f_n)\}$ to denote an *r*-itemset.

Example 2. For example, {(*MS*, *very-low*), (*FS*, *low*)} is an *r*-itemset.

Definition 3 (Fuzzification). Suppose we have a universe of discourse *X* in a *quantitative domain*, where each element *x* belongs to *X*. Then, fuzzy set *F* is characterized by a membership function $m_F(x)$, which maps *x* to a membership degree in interval [0, 1].

Example 3. Assume that we have five membership functions, $S_{very\text{-}low}$, S_{low}, S_{middle}, S_{high} and $S_{very\text{-}high}$ for the scores. From these five membership functions, we know that $S_{very\text{-}low}(65) = 1$, $S_{low}(79) = 0.9$, $S_{middle}(81) = 0.9$, and $S_{high}(89) = 0.9$.

Table 1
Educational data set.

TID	Itemsets
1	(*MS*, 60), (*FS*, 79), (*QS*, 67), (*HS*, 62), (*AS*, 86)
2	(*MS*, 77), (*FS*, 79), (*QS*, 80), (*HS*, 84), (*AS*, 86)
3	(*MS*, 78), (*FS*, 78), (*QS*, 78), (*HS*, 78), (*AS*, 85)
4	(*MS*, 82), (*FS*, 83), (*QS*, 84), (*HS*, 80), (*AS*, 90)
5	(*MS*, 91), (*FS*, 92), (*QS*, 93), (*HS*, 95), (*AS*, 96)

MS, mid-term score; *FS*, final-term score; *QS*, quiz score; HS, homework score; *AS*, attendance score.

$$S_{very\text{-}low}(s)=\begin{cases}1, & \text{if } s\leqslant 65\\ \frac{70-s}{70-65}, & \text{if } 65\leqslant s\leqslant 70\end{cases}; \quad (1)$$

$$S_{low}(s)=\begin{cases}\frac{s-65}{70-65}, & \text{if } 65\leqslant s\leqslant 70\\ 1, & \text{if } s=70\\ \frac{80-s}{80-70}, & \text{if } 70\leqslant s\leqslant 80\end{cases}; \quad (2)$$

$$S_{middle}(s)=\begin{cases}\frac{s-70}{80-70}, & \text{if } 70\leqslant s\leqslant 80\\ 1, & \text{if } s=80\\ \frac{90-s}{90-80}, & \text{if } 80\leqslant s\leqslant 90\end{cases}; \quad (3)$$

$$S_{high}(s)=\begin{cases}\frac{s-80}{90-80}, & \text{if } 80\leqslant s\leqslant 90\\ 1, & \text{if } s=90\\ \frac{95-s}{95-90}, & \text{if } 90\leqslant s\leqslant 95\end{cases}; \quad (4)$$

$$S_{very\text{-}high}(s)=\begin{cases}\frac{s-90}{95-90}, & \text{if } 90\leqslant s\leqslant 95\\ 1, & \text{if } 95\leqslant s\end{cases}. \quad (5)$$

By applying the above membership functions, we find that the number value 79 is 0.1/*low* + 0.9/*middle* (+ denote union). Since a number value (79) can be mapped to multiple *linguistic* variables, that is, *low* and *middle*, we define the *linguistic* variable which holds the maximum degree to represent the number value. In this study it is suggested that the degree of the number value (79) that matches the *linguistic* variable (*middle*) is 0.9.

Both superior and inferior learning situations are discovered during the mining of fuzzy rare itemsets for the purpose of discovering learning problems, being both abnormal and rare. In other words, using only the threshold *sup*, one cannot tell the difference between superior and inferior learning situations. Therefore, another threshold *rnk* must be added. This is an index used for ranking the student's score in a specific course. With the threshold *rnk*, we can identify which learning situation is superior or inferior within the fuzzy rare itemsets. The higher the *rnk* value, the more superior the learning situation.

Definition 4. Assume that we have a *number s*-item $a_i=(it_i,s_i)$, a *linguistic r*-item $b_j=(ic_j,f_j)$, and a membership function (FS_{f_j}), where $FS_{f_j}(s_i)$ denotes the membership degree to which s_i belongs to f_j. Then, the degree to which a_i matches b_j, $sup(a_i,b_j)$ can be given as follows:

$$sup(a_i,b_j)=\begin{cases}FS_{f_j}(s_i), & \text{if } it_i=ic_j\\ 0, & \text{otherwise}\end{cases}.$$

When using the same scale to evaluate the rank for each score for each course, the threshold, *rnk*, has to be standardized. Given all students' scores for a specific course, we can calculate the max-score ($s_{\max}$) and min-score ($s_{\min}$). Now, $rnk(a_i,b_j)$ can be given by

$$rnk(a_i,b_j)=\begin{cases}\frac{s_i-s_{\min}}{s_{\max}-s_{\min}}, & \text{if } it_i=ic_j\\ 0, & \text{otherwise}\end{cases}.$$

Example 4. Suppose we have a *number s*-item $a_1=(it_1,67)$, a *linguistic r*-item $b_1=(ic_1,low)$, and the membership function (S_{low}), as shown in Example 3. Now, the degree $sup(a_1,b_1)=sup((it_1,67),(ic_1,low))=0.6$ and the degree $rnk(a_1,b_1)=rnk((it_1,67)=(67-67)/(93-67)=0$.

Definition 5. A *rule r*-item can be a *linguistic r*-item. For simplicity, we use $b_i=(ic_i,f_i)$ to denote a *rule r*-item. A *rule r*-itemset B is a set of *rule r*-items, that is, $B=\{(ic_1,f_1),(ic_2,f_2),\ldots,(ic_n,f_n)\}$ to denote a *rule r*-itemset.

Example 5. For example, {(*MS*, *very-low*), (*QS*, *very-low*), ((*HS*, *very-low*)} is a *rule r*-itemset.

Definition 6. Let an *s*-itemset be a set of *s*-items. Assume that we have an *s*-itemset $A=\{(it_1,s_1),(it_2,s_2),\ldots,(it_m,s_m)\}$, where $a_i=(it_i,s_i)$ could be a *number s*-item. Also assume that we have a *rule r*-itemset $B=\{(ic_1,f_1),(ic_2,f_2),\ldots,(ic_n,f_n)\}$, where $b_j=(ic_j,f_j)$ is a *rule r*-item. If we can find $a_{i_1}\leqslant a_{i_2}\leqslant\cdots\leqslant a_{i_n}$ in A, such that $\sup(a_{i_j},b_j)>0$, then $sup(A,B)$ and $rnk(A,B)$ can be defined as follows:

$$sup(A,B)=\mathrm{Min}_{j=1}^{n}sup(a_{i_j},b_j),$$
$$rnk(A,B)=\mathrm{Max}_{j=1}^{n}rnk(a_{i_j},b_j).$$

Example 6. Suppose we have an *s*-itemset $A=\{(MS,60),(FS,79),(QS,67),(HS,62),(AS,86)\}$, a *rule r*-itemset $B=\{(MS,$ *very-low*$),(QS,$ *very-low*$),(HS,$ *very-low*$)\}$. The membership functions are shown in Example 3. If $it_i=ic_i$ for $1\leqslant i\leqslant 5$, then the degree $sup(A,B)=\min(1.0,0.6,1.0)=0.6$ and $rnk(A,B)=\max(0,0,0)=0$.

Definition 7. Assume that we have a database D consisting of a set of transactions, where the *sid*-th transaction in D can be represented as an *s*-itemset $A_{sid}=\{(it_1,s_1),(it_2,s_2),\ldots,(it_m,s_m)\}$. Let $B=\{(ic_1,f_1),(ic_2,f_2),\ldots,(ic_n,f_n)\}$ be a *rule r*-itemset. Then, the support of B occurring in D, $sup_D(B)$ can be defined as follows:

$$sup_D(B)=\left(\sum_{sid=1}^{|D|}sup(A_{sid},B)\right)\Big/|D|,$$

where $|D|$ is the total number of transactions in database D.

The rank of B in database D, denoted as $rnk_D(B)$, can be defined as follows:

$$rnk_D(B)=\left(\sum_{sid=1}^{|D_B|}rnk(A_{sid},B)\right)\Big/|D_B|,$$

where $|D_B|$ is the subset of transactions in database D with itemset B.

Example 7. Suppose we have a database D containing 5 educational records, as shown in Table 1, and a *rule r*-itemset $B=\{(MS,$ *very-low*$),(QS,$ *very-low*$),(HS,$ *very-low*$)\}$. Now, the degree $sup_D(B)=(\min(1.0,0.6,1.0)+\min(0,0,0)+\min(0,0,0)+\min(0,0,0)+\min(0,0,0))/5=0.6/5=0.12$, and the degree $rnk_D(B)=\max(0,0,0)/1=0/1=0$.

Definition 8. Given a user-specified threshold σ_s, a *rule r*-itemset B is rare if $sup_D(B)$ is no larger than σ_s. Let B be a rare *rule r*-itemset, where $B=X\cup Y$ and $X\cap Y=\phi$. Then, the confidence of rule $X\Rightarrow Y$, denoted as $conf(X\Rightarrow Y)$, is defined as $sup_D(B)/sup_D(X)$. Given a confidence threshold σ_c, if $conf(X\Rightarrow Y)\geqslant\sigma_c, X\Rightarrow Y$ holds in database D.

Example 8. According to Definitions 3 and 7, only *rule r*-items may appear in both sides of the generated fuzzy rules. Therefore, we may have rules like (*MS.very-low*) → (*HS.very-low*), (*HS.very-low*) → (*QS.very-low*), and (*QS.very-low*) → (*HS.very-low*). We never, however, generate rules like (*MS.very-low*) → (*MS*.63).

4. Algorithm for mining fuzzy rare itemsets from educational data

In this section, we introduce an Apriori-like algorithm, the Fuzzy Apriori Rare Itemsets Mining (*FARIM*) algorithm, to discover

fuzzy rare itemsets from educational data. The *FARIM* algorithm was developed by modifying the well-known Apriori algorithm [2,3].

We now introduce a new algorithm for mining fuzzy rare itemsets from educational data. The algorithm is outlined in Fig. 2 (for brevity the proof is given in Appendix A). Although the basic structure of the *FARIM* is similar to that of the Apriori algorithm, they are different in the following respects:

(1) Data type: the Apriori algorithm is designed only for handling categorical data. The *FARIM* algorithm, however, is developed for handling the quantitative educational data.
(2) Membership functions: in the Apriori algorithm, an item can have only a 100% or 0% match with another item. A membership function is not needed to measure the degree of membership between items. To deal with the partial membership relationships that exist in raw data, the *FARIM* algorithm uses the membership functions described in Section 3 to calculate the membership degree between items.
(3) Itemset type: the Apriori algorithm is designed only for discovering frequent itemsets. The *FARIM* algorithm, however, is developed for discovering fuzzy rare itemsets that may appear in educational data.
(4) Counting candidates: in the Apriori algorithm, an itemset is either completely contained in a transaction or not. In the *FARIM* algorithm, an itemset can be partly contained in a transaction. As a result, the degree that a transaction contains an itemset is assigned a value between 0 and 1, instead of either 1 or 0.

Besides the differences mentioned above, there is one more distinction that makes the structure of our algorithm different from the Apriori algorithm.

Definition 9 (*Anti-monotone constraints*). A constraint ζ is anti-monotone, which means that if it holds for an itemset S, then it holds for any subset of S.

Property 1. Let the constraint that $sup_D(B)$ must be no less than minimum support (0) and that $rnk_{DB}(B)$ must be no larger than σ_{rnk} be called the support constraint and the rank constraint, respectively. The support constraint satisfies the anti-monotone property, while the *rank* constraint does not.

Proof. It is obvious that the support constraint satisfies the anti-monotone property. Here, we use a counter example to show why the *rank* constraint does not satisfy the anti-monotone property. Assume that we have a data set composed of two transactions, as shown in Table 2. The *rank* of item X is $(0.3 + 0.4)/2 = 0.35$, but the *rank* of itemset XY is 0.3. If $\sigma_{rank} = 0.33$, then XY satisfies the *rank* constraint, but its subset X does not.

Property 1 indicates that we can use the minimum support constraint to develop the algorithm, similar to the traditional Apriori algorithm. Furthermore, we can use the maximum support as a threshold to determine *rare* r-itemsets. Since the *rank* constraint does not satisfy the anti-monotone property, after finding the rare r-itemsets in each phase, we must further apply the rank constraint to prune the rare r-itemsets and then determine *low-rank* rare r-itemsets.

The proposed *FARIM* algorithm has three phases. In the first phase, the membership functions in Section 3 are applied to transform the original database into a new database. After the transformation, a transaction in the new database stores the support of every r-item in the corresponding transaction of the original database. In the second phase, a level-wise approach is used to iteratively generate candidate r-itemsets of k items C_k, and then

Input: A database D; membership functions (FS_{f_j}); a predefined minimum support $\sigma^R_{\min\sup}$; a predefined maximum support $\sigma^R_{\max\sup}$; a predefined maximum rank $\sigma^R_{\max rnk}$; a predefined minimum confidence λ.
Output: A set of fuzzy association rules
Method:
// Phase 1 Call the *Sup_Transform* Subroutine
(1). For each transaction
  Transform each s-item data into r-items;
  Store these results as a new transaction in new database D^T.
// Phase 2 Call the *RareItemsets_gen* Subroutine
(1). Calculate the support for each r-item $ic_{j,r}$.
(2). Check whether the support for each r-item ic_j is no less than the minimum support, $\sigma^R_{\min\sup}$. If it is, assign it to the set of frequent one-itemsets (L_1).
(3). Check whether the support of each r-item ic_j is no larger than the maximum support, $\sigma^R_{\max\sup}$. If it is, assign it to the set of *rare* one-itemsets (R_1).
(4). Check whether the rank of each r-itemset in R_1 is no larger than the maximum rank $\sigma^R_{\max rnk}$. If it is, assign it to the set of *low-rank rare* one-itemsets (R^{lr}_1).
(5). Generate candidate set C_{k+1} from L_k.
(6). Compute the supports for all r-itemsets in C_{k+1} and then determine R_{k+1} and L_{k+1}.
(7). Compute the ranks of all r-itemsets in L_{k+1} and then determine R^{lr}_{k+1}.
(8). If L_{k+1} is null, go to phase 3; otherwise, set $k = k + 1$ and repeat steps (3)–(7).
// Phase 3 Call the *FAR_gen* Subroutine
(1). Generate fuzzy association rules from all rare *rule* r-itemsets.

Fig. 2. *FARIM* algorithm.

Table 2
Ranks of the items.

TID	X	Y	Z
1	0.3	0.3	0.1
2	0.4		0.3

find rare r-itemsets of k items R_k. Furthermore, with the help of the threshold named *rnk*, we can determine low rank rare r-itemsets of k items R_k^{lr}. To go to the next level, a candidate set C_{k+1} is generated from L_k and the same procedure repeated. In the final phase, fuzzy association rules are generated from the low rank r-itemsets R_k^{lr} obtained in the second phase. The three subroutines for each phase are explained in detail below.

As mentioned above, educational data are quantitative data. Accordingly, we need to use the membership functions to calculate their supports. Fig. 3 shows the pseudocode for the *Sup_Transform* subroutine, which is used to calculate the support. In steps 1-5, we check every s-item a_i in each transaction, and generate the corresponding r-items. Finally, all s-item a_i will be transformed into r-items of the form (b_j, μ_j, r_j), where b_j is an r-item; μ_j is its support; and r_j is its rank. Step 3 is used to calculate its support. With the help of the membership functions, the *Sup_Transform* subroutine can transform the original dataset into a new form D^T. In the next section, we will introduce a new method, the *RareItemsets_gen* subroutine, for mining low-rank rare r-itemsets from database D^T.

Unlike the Apriori algorithm, which calculates the counts of candidate itemsets by adding either a one or a zero, depending on whether that particular itemset appears in the transaction or not, the *RareItemsets_gen* subroutine can add a fractional value to the counts of the itemsets. In addition, the minimum support is set to zero, and then all *rare* r-itemsets are discovered. Fig. 4 shows the pseudocode for the *RareItemsets_gen* subroutine. Step 1 finds the frequent 1-itemsets L_1. In steps 2–10, L_{k-1} is used to generate candidates C_k in order to find L_k. The *apriori_gen* subroutine generates the candidates. Unlike the *apriori_gen* subroutine in the Apriori algorithm [3] calculating *count* only, the *apriori_gen* subroutine used in this study will calculate both *count* and *rank*. The downward closure property is used to eliminate those that have a non-frequent subset (step 3). Once all the candidates have been generated, the database is scanned (step 4). For each transaction, a *subset* function is used to find all subsets of the transaction that are candidates (step 5), and the count and rank for each of these candidates are accumulated (steps 6, 7 and 8). Finally, all those candidates satisfying the minimum support constraint constitute the set of frequent r-itemsets (L_k) (steps 13). After filtering out *non-rare* r-itemsets from L_k, we obtain *rare* r-itemsets (R_k) (steps 14) with supports less than the maximum support constraint. Furthermore, the *rnk* constraint is applied to determine *low-rank rare* r-itemsets (R_k^{lr}) (steps 15). Afterwards, these patterns are used to generate the *Fuzzy association rules (FAR)*. In the next section, the *FAR_gen* subroutine to generate fuzzy association rules from *low-rank rare* r-itemsets (R_k^{lr}) will be introduced.

Finally, like the Apriori algorithm, fuzzy association rules are generated from *low-rank rare* r-itemsets (R_k^{lr}) obtained in the second phase. Fig. 5 shows the pseudocode for the *FAR_gen* subroutine. Obviously, the procedure can generate all the *FAR*s satisfying Definition 8.

5. An example using the proposed algorithm

An example is given to illustrate the proposed data mining algorithm using the dataset shown in Table 1.

Step 1. Assume that we have five membership functions (FS_{f_j}) as shown in expressions (1), (2), (3), (4) and (5). Although every s-item (it_i, s_i) in each transaction of D can be mapped to a set of r-items with multiple supports, this study holds max-support to represent the s-item. Further, by consolidating the rank of every r-item into two values, support and rank, we can build a temporary database D^T as shown in Table 3.

Step 2.1. Calculate the support for each r-item stored in database D^T and check whether the support is larger than the minimum support (0). If it is, put it in L_1. Now, filter out *non-rare* r-itemsets from L_1 to obtain the *rare* r-itemsets (R_1) with supports less than or equal to the maximum support constraint ($\sigma^R_{\max sup}$). Finally, calculate the rank and check whether the rank of each itemset in R_1 is less than or equal to the maximum rank $\sigma^R_{\max rnk}$. If it is, put it into the set of R_1^{lr}. For example, let us set $\sigma^R_{\max sup}$ to 0.2 and $\sigma^R_{\max rnk}$ to 0.2. Now we have R_1 and R_1^{lr}, as shown in Table 4.

Step 2.2. We now generate candidate set C_2 from L_1. For example, C_2 is obtained as follows: (*MS.very-low*, *MS.high*), (*MS.very-low*, *FS.high*), (*MS.very-low*, *QS.very-low*), (*MS.very-low*, *QS.very-high*), ..., and (*AS.middle*, *AS.very-high*). After computing the supports, we can determine L_2 and then filter out *non-rare* r-itemsets from L_2 to obtain *rare* r-itemsets (R_2), as shown in Table 5. Next, filter out *high-rank rare* r-itemsets from R_2 to obtain only *low-rank rare* r-itemsets (R_2^{lr}), as shown in Table 6.

Step 2.3. Since L_2 is not null, repeat the previous steps to find R_k^{lr}. Finally, we find C_6 is empty after pruning; therefore, we stop the iterations.

Subroutine: *Sup_Transform* Subroutine. Transform every s-item data (it_i, s_i) in each transaction of D into a set of (ic_j, μ_j, r_j), where ic_j is an r-item, μ_j is its support and r_j is its rank.
Input: A database D; a membership function FS_{f_j}.
Output: D^T, a new database where each transaction contains a set of (b_j, μ_j, r_j)
(1) for each transaction $t \in D$ {
(2) for each s-item $a_i=(it_i, s_i)$ {
// the following procedure does not generate any r-item if its support is zero//
(3) if $a_i \in$ *number* s-item, then for each linguistic term f_j
create an r-item $b_j= (ic_i, f_j)$ with *sup* and *rnk* as defined in Definition 4. }
(4) store the results as a transaction in the new database D^T;}
(5) return D^T;

Fig. 3. *Sup_Transform* function.

Subroutine: *RareItemsets_gen* Subroutine. Find *low-rank rare r*-itemsets using an iterative level-wise approach based on candidate generation.

Input: Database D^T; the minimum support $\sigma^R_{\min sup}$; the maximum support $\sigma^R_{\max sup}$ and the maximum rank $\sigma^R_{\max rnk}$.

Output: R^{lr}_k, *low-rank rare r*-itemsets in D^T.

Method:

(1) L_1 = find_frequent_1-itemsets(D^T);
(2) for (k=2; $L_{k-1}\neq\emptyset$; k++) {
(3) C_k = apriori_gen(L_{k-1}, $\sigma^R_{\min sup}$, $\sigma^R_{\max sup}$, $\sigma^R_{\max rnk}$);
(4) for each transactions $t\in D^T$ { //Scan D to compute counts
(5) C^t=subset(C_k, t); //get the subsets of t that are candidates;
(6) for each candidate $c \in C^t$ {
(7) $c.count=c.count+Min^k_{j=1}sup_j$;
(8) $c.rnkcount=c.rnkcount+Max^k_{j=1}rnk_j$;
(9) }
(10) }
(11) $c.sup=(c.count/|D|)$;
//where $|D|$ denote the transaction numbers of the database D.
(12) $c.rnk=(c.rnkcount/|D_B|)$;
//where $|D_B|$ denote the transaction numbers of the itemsets B in the database D.
(13) $L_k=\{c\in C_t \mid c.sup>\sigma^R_{\min sup}\}$;
(14) $R_k=\{c\in L_k \mid c.sup\leq\sigma^R_{\max sup}\}$;
(15) $R^{lr}_k=\{c\in R_k \mid c.rnk\leq\sigma^R_{\max rnk}\}$;
(16) return $L=\cup_k L_k$};

Fig. 4. *RareItemsets_gen* function.

Subroutine: *FAR_gen* Subroutine. Generate all fuzzy association rules

Input: R^{lr}_k, *low-rank rare r*-itemsets in D^T; a predefined minimum confidence λ.

Output: *FAR*, Fuzzy association rules in D^T.

Method:

(1) For each *low-rank rare r*-itemsets $B=X\cup Y$, where $X\cap Y=\phi$
(2) For every subset X of B
(3) If the confidence of rule $X\Rightarrow Y$ is no less than the minimum confidence λ
(4) then output the rule
(5) return *FAR*

Fig. 5. *FAR_gen* function.

Table 3
Constructed temporary set D^T.

TID	*MS*	*FS*	*QS*	*HS*	*AS*
1	(*very-low*, 1, 0)	(*middle*, 0.9, 0.07)	(*very-low*, 0.6, 0)	(*very-low*, 1, 0)	(*high*, 0.6, 0.09)
2	(*middle*, 0.7, 0.55)	(*middle*, 0.9, 0.07)	(*middle*, 1, 0.5)	(*middle*, 0.6, 0.67)	(*high*, 0.6, 0.09)
3	(*middle*, 0.8, 0.58)	(*middle*, 0.8, 0)	(*middle*, 0.8, 0.42)	(*middle*, 0.8, 0.48)	(*middle*, 0.5, 0)
4	(*middle*, 0.8, 0.71)	(*middle*, 0.7, 0.36)	(*middle*, 0.6, 0.65)	(*middle*, 1, 0.55)	(*high*, 1, 0.45)
5	(*high*, 0.8, 1)	(*high*, 0.6, 1)	(*very-high*, 0.6, 1)	(*very-high*, 1, 1)	(*very-high*, 1, 1)

Table 4
R_1 and R_1^{lr}.

Rare r-itemsets in R_1			*Low-rank-rare r*-itemsets in R_1^{lr}		
Itemsets	Support	Rank	Itemsets	Support	Rank
MS.very-low	0.20	0.00	*MS.very-low*	0.20	0.00
MS.high	0.16	1.00	*QS.very-low*	0.12	0.00
FS.high	0.12	1.00	*HS.very-low*	0.20	0.00
QS.very-low	0.12	0.00	*AS.middle*	0.10	0.00
QS.very-high	0.12	1.00			
HS.very-low	0.20	0.00			
HS.very-high	0.20	1.00			
AS.middle	0.10	0.00			
AS.very-high	0.20	1.00			

Table 5
R_2 for this example.

Rare r-itemsets (R_2)	*Rare r*-itemsets (R_2)	*Rare r*-itemsets (R_2)
(*MS.very-low*, *FS.middle*)	(*MS.high*, *AS.very-high*)	(*QS.very-low*, *AS.high*)
(*MS.very-low*, *QS.very-low*)	(*FS.middle*, *QS.very-low*)	(*QS.middle*, *AS.middle*)
(*MS.very-low*, *HS.very-low*)	(*FS.middle*, *HS.very-low*)	(*QS.very-high*, *HS.very-high*)
(*MS.very-low*, *AS.high*)	(*FS.middle*, *AS.middle*)	(*QS.very-high*, *AS.very-high*)
(*MS.middle*, *AS.middle*)	(*FS.high*, *QS.very-high*)	(*HS.very-low*, *AS.high*)
(*MS.high*, *FS.high*)	(*FS.high*, *HS.very-high*)	(*HS.middle*, *AS.middle*)
(*MS.high*, *QS.very-high*)	(*FS.high*, *AS.very-high*)	(*HS.very-high*, *AS.very-high*)
(*MS.high*, *HS.very-high*)	(*QS.very-low*, *HS.very-low*)	

Step 3. Construct the *FAR*s from all *low-rank rare r*-itemsets which are rare and *low-rank*.
Generate *FAR*s from *low-rank rare r*-itemsets $R_2^{lr}, R_3^{lr}, R_4^{lr}$ and R_5^{lr}. For example, let us set λ to 0.9. Now we have 9 *FAR*s from *low-rank rare r*-itemsets R_2^{lr}, as shown in Table 7.

6. Experimental results

Experiments were conducted to evaluate the approach and the performance of the proposed algorithm. The survey data consisted of the scores of undergraduate students in the College of Management during the 2010 semester. The data were obtained from the Computer Center at University CTUST. A total of 62875 items of survey data were collected. Each transaction gave information about the student's learning performance represented by scores. The algorithms were implemented using the Sun Java language (J2SDK 1.3.1). Testing was carried out on a Notebook with a single Intel Centrino 1400 MHz processor and 512 MB main memory using the Windows XP operating system. Neither multi-threading technology nor parallel computing skills were used for program implementation. A senior faculty from our department was invited to set the values of the five membership functions. According to his suggestion, five degrees were set, very-low, low, middle, high, and very-high, corresponding to the core score values, 65, 70, 80, 90, and 95, respectively. The five membership functions named $S_{very\text{-}low}$, S_{low}, S_{middle}, S_{high}, and $S_{very\text{-}high}$ are set as shown in Section 3. The senior faculty member was also invited to set the values of the four thresholds named minimum support, maximum support, maximum rank and minimum confidence, respectively.

Table 7
Fuzzy association rules generated from *low-rank rare r*-itemsets R_2^{lr}.

No.	Rule
1	If *MS.very-low* then *FS.middle*; (confidence = 90%; rank = 0.07)
2	If *QS.very-low* then *MS.very-low*; (confidence = 100%; rank = 0.00)
3	If *MS.very-low* then *HS.very-low*; (confidence = 100%; rank = 0.00)
4	If *HS.very-low* then *MS.very-low*; (confidence = 100%; rank = 0.00)
5	If *QS.very-low* then *FS.middle*; (confidence = 100%; rank = 0.07)
6	If *HS.very-low* then *FS.middle*; (confidence = 90%; rank = 0.07)
7	If *AS.middle* then *FS.middle*; (confidence = 100%; rank = 0.00)
8	If *QS.very-low* then *HS.very-low*; (confidence = 100%; rank = 0.00)
9	If *QS.very-low* then *AS.high*; (confidence = 100%; rank = 0.09)

Before detailing the experiments, the differences between the three approaches, Fuzzy-Apriori [12], Fuzzy-Apriori-Rare and *FARIM*, are summarized as shown in Table 8. Since all three approaches are based on the Apriori algorithm, they are similar in structure. The Fuzzy-Apriori [12] approach is used to discover fuzzy frequent itemsets. The other two approaches, Fuzzy-Apriori-Rare and the *FARIM* proposed in this study, are used to discover fuzzy rare itemsets.

Although the basic structure of the three approaches, Fuzzy-Apriori, Fuzzy-Apriori-Rare and *FARIM* are based on the Apriori algorithm, they differ in the following respects: aim, thresholds and learning behavior types. As shown in Table 8, the Fuzzy-Apriori approach which is designed for discovering fuzzy frequent itemsets, uses only two thresholds, *minimum-support* and *minimum-confidence*. Since normal learning behaviors frequently appear as fuzzy frequent itemsets, most normal learning behaviors can be found with the Fuzzy-Apriori approach. If we want to

Table 8
Comparison of the three approaches.

Algorithm	Aim	Thresholds	Learning behavior
Fuzzy-Apriori [12]	Fuzzy frequent itemsets	**Minimum support** **Minimum confidence**	Normal learning behaviors
Fuzzy-Apriori-Rare (modified from Fuzzy-Apriori by the authors)	Fuzzy rare itemsets	**Minimum support** **Minimum confidence** Maximum support	Part of normal learning behaviors and all negative learning behaviors
FARIM (modified from Fuzzy-Apriori-Rare by the authors)	Fuzzy specific rare itemsets	**Minimum support** **Minimum confidence** Maximum support Maximum rank	All negative learning behaviors only

Table 6
R_2^{lr} for this example.

Itemsets	Support	Rank	Itemsets	Support	Rank
(*MS.very-low*, *FS.middle*)	0.18	0.07	(*FS.middle*, *HS.very-low*)	0.18	0.07
(*MS.very-low*, *QS.very-low*)	0.12	0.00	(*FS.middle*, *AS.middle*)	0.10	0.00
(*MS.very-low*, *HS.very-low*)	0.20	0.00	(*QS.very-low*, *HS.very-low*)	0.12	0.00
(*MS.very-low*, *AS.high*)	0.12	0.09	(*QS.very-low*, *AS.high*)	0.12	0.09
(*FS.middle*, *QS.very-low*)	0.12	0.07	(*HS.very-low*, *AS.high*)	0.12	0.09

discover abnormal learning behaviors which appear infrequently, the *minimum-support* must be set low to obtain these fuzzy rare itemsets. An extra *maximum-support* threshold is applied to filter out non-rare itemsets. Thus, Fuzzy-Apriori-Rare is designed by this idea to discover fuzzy rare itemsets. However, Fuzzy-Apriori-Rare can discover not only abnormal learning behaviors, but also some normal learning behaviors. Therefore, we can not identify which fuzzy rare itemsets represent negative learning behaviors from fuzzy rare itemsets generated by Fuzzy-Apriori-Rare approach. To address this drawback of Fuzzy-Apriori-Rare approach, the proposed *FARIM* uses the extra measure, *maximum-rank*, to obtain only fuzzy specific rare itemsets which represent negative learning behaviors.

The major difference between Fuzzy-Apriori and Fuzzy-Apriori-Rare is that the Fuzzy-Apriori-Rare applies the maximum support ($\sigma^R_{\max sup}$) to filter out fuzzy non-rare itemsets. Furthermore, the *FARIM* approach uses the maximum rank ($\sigma^R_{\max rnk}$) to filter fuzzy specific rare itemsets (*low-rank rare r*-itemsets). Therefore, the major difference between Fuzzy-Apriori-Rare and *FARIM* is that *FARIM* uses the measure named maximum rank ($\sigma^R_{\max rnk}$) to discover fuzzy specific rare itemsets.

There are three experiments detailed in this section. In the first, the Fuzzy-Apriori-Rare algorithm, modified from the traditional Fuzzy-Apriori algorithm [12], is applied to discover fuzzy rare itemsets for which the supports are not larger than maximum support. Changes in the run-time for the Fuzzy-Apriori-Rare algorithm with the minimum support value and the size of the database are investigated. In the second experiment, the *FARIM* algorithm is used to discover fuzzy rare itemsets in advance. The performance of the proposed *FARIM* algorithm is compared with that of the Fuzzy-Apriori-Rare algorithm, which uses the Min operator to infer the supports of the itemsets. Finally, in the third experiment, the *FARIM* algorithm is applied to discover rules from real world educational data. Unlike the Fuzzy-Apriori-Rare algorithm, which discovers rules from fuzzy rare itemsets, the proposed *FARIM* algorithm can discover interesting rules only from *low-rank* fuzzy rare itemsets.

In the first experiment, changes in the run-time are investigated as we vary the minimum support value and database size. The database size is set at 62875, the maximum rank at 0.53 and the minimum support varied. The results obtained with the Fuzzy-Apriori-Rare approach are shown in Fig. 6. It is apparent that the run-time increases with a decrease in the minimum support value. This is especially true when the minimum support becomes very small in which case the run time increases sharply. These results concur with the results obtained with previous association mining algorithms [12]. Next, the maximum support is set to 0.2, and the number of transactions varied. It can be seen in Fig. 7 that the run-time increases linearly with respect to the database size. This linear relationship indicates that the Fuzzy-Apriori-Rare approach has a good scalability.

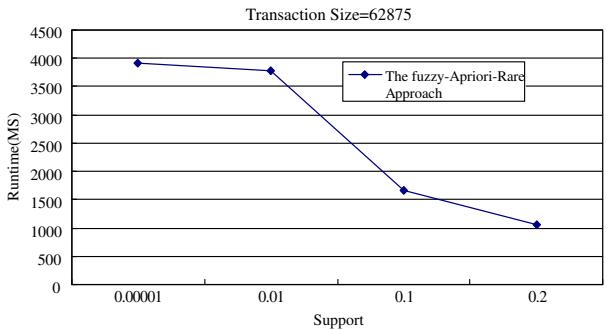

Fig. 6. Run-time vs. minimum support.

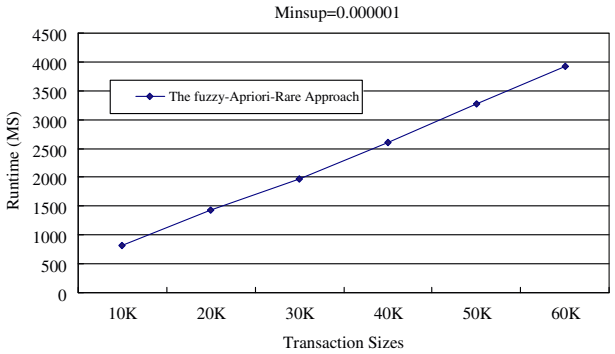

Fig. 7. Run-time vs. database size.

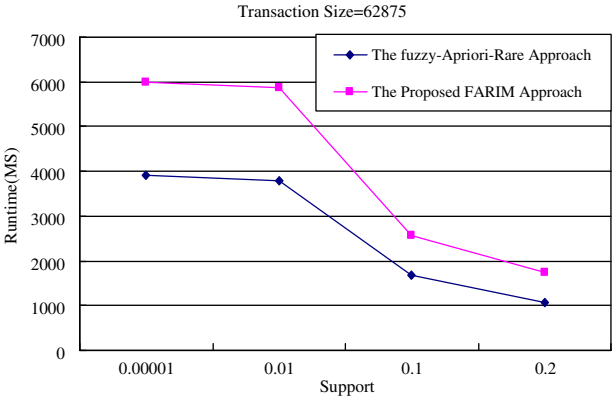

Fig. 8. Run-time vs. minimum support.

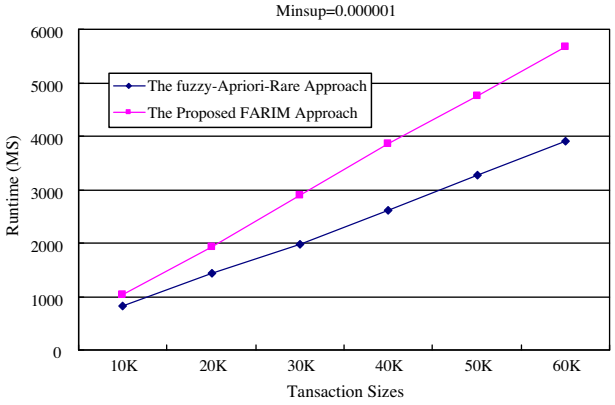

Fig. 9. Run-time vs. database size.

The second experiment is aimed at studying differences in performance between the Fuzzy-Apriori-Rare algorithm and the proposed *FARIM* algorithm. Note that in the Fuzzy-Apriori-Rare algorithm the rank factor is not considered, while it is considered in the proposed *FARIM*. To make comparison possible, we need to generate a specific transaction set for the Fuzzy-Apriori-Rare algorithm. Since the *Sup_Transform* function used in this work could generate a new database D^T, where each transaction contains a set of (b_j, μ_j, r_j), the rank part (r_j) from each transaction set of (b_j, μ_j, r_j) is eliminated. A new transaction set of (b_j, μ_j) is obtained, which can be used directly by the Fuzzy-Apriori-Rare algorithm. It is now possible to compare the performance of the Fuzzy-Apriori-Rare algorithm and the proposed *FARIM* algorithm.

In the first test, the size of the database is set at 62875 and the minimum support varied from 0.000001 to 0.20. The results in

Fig. 8 show that the Fuzzy-Apriori-Rare algorithm performs slightly better than the proposed *FARIM* algorithm. This result is quite reasonable, because the proposed *FARIM* algorithm includes the extra *rank* factor which requires additional computation.

In the second test, the minimum support is set to 0.000001. Fig. 9 indicates that the Fuzzy-Apriori-Rare algorithm has a better run-time than the *FARIM* algorithm. The reason for this result is the same as that stated for the first test.

In the third test, the size of the database is set to 62875 and the maximum support vaired from 0.1 to 0.4. Table 9 shows the accumulative number of patterns in R_1, R_2, R_3, R_4 and R_5 for different maximum supports $\sigma^R_{\max sup}$. Obviously, the Fuzzy-Apriori-Rare algorithm generates many more R_1, R_2, R_3, R_4 and R_5 patterns than the proposed *FARIM* algorithm. The results indicate that the many *rare* patterns generated by the Fuzzy-Apriori-Rare algorithm contain both low rank and high rank patterns, rather than low rank patterns only. With the new index *rank*, the proposed *FARIM* approach discovers only the *low-rank rare* patterns from educational data.

The third experiment demonstrated the number of patterns for different rank thresholds. The size of the database is set to 62875, the maximum support thresholds $\sigma^R_{\max sup}$ to 0.20 and the maximum rank varied from 0.53 to 1.00. Table 10 shows the accumulative numbers of patterns in R_1, R_2, R_3, R_4 and R_5 for different maximum rank thresholds $\sigma^R_{\max rnk}$. It is apparent that the number of patterns increases with an increase in maximum rank value. This is especially true when the maximum rank becomes very large.

Table 9
Number of fuzzy rare itemsets generated by Fuzzy-Apriori-Rare and *FARIM*.

Maxsup	Fuzzy-Apriori-Rare				*FARIM*			
	0.10	0.20	0.30	0.40	0.10	0.20	0.30	0.40
R1	2	18	24	24	0	0	3	4
R2	233	238	239	239	3	3	4	4
R3	1052	1053	1053	1053	1	1	1	1
R4	2013	2013	2013	2013	0	0	0	0
R5	1085	1085	1085	1085	0	0	0	0
Total	4385	4407	4414	4414	4	4	8	9

Table 10
Number of fuzzy rare itemsets generated by *FARIM* with different *max-rnk*.

Maxrnk	0.53	0.60	0.70	0.80	0.90	1.00
R1	0	1	1	6	13	18
R2	3	9	13	47	147	238
R3	1	10	18	115	534	1053
R4	0	5	10	117	814	2013
R5	0	1	2	43	376	1085
Total	4	26	44	328	1884	4407

Table 11
Low-rank fuzzy rare itemsets obtained in this experiment.

Itemsets	Support (%)	Rank (%)
(*MS.very-low*, *HS.very-low*)	7.97	52.46
(*FS.very-low*, *HS.very-low*)	9.04	52.88
(*QS.very-low*, *HS.very-low*)	7.45	52.51
(*MS.very-low*, *QS.very-low*, *HS.very-low*)	7.34	52.59

Obviously, with the new index *rank*, proposed in this work, we could eliminate many high-rank patterns, thereby obtaining *low-rank* ones from educational data. The above results show that the proposed *FARIM* is able to discover interesting rules containing only *low-rank* fuzzy rare itemsets.

The difference between the two approaches is shown by comparing the number of fuzzy rare itemsets discovered with each approach. With the maximum support values $\sigma^R_{\max sup} = 0.2$ and the maximum rank value $\sigma^R_{\max rnk} = 0.53$, the proposed *FARIM* approach discovered only 4 fuzzy rare itemsets. However, with the same support values $\sigma^R_{\max sup} = 0.2$, and without the rank threshold, the Fuzzy-Apriori-Rare approach discovered 4407 fuzzy rare itemsets, as shown in Table 9. The Fuzzy-Apriori-Rare approach generates 235 R^{lr}_2 extra patterns with a rank larger than 0.53. Without the rank threshold, the Fuzzy-Apriori-Rare approach will generate many high-rank patterns. Furthermore, please note that when we set maximum rank to 1.0, the number of patterns in R_1, R_2, R_3, R_4 and R_5 in Table 10 are the same as the number of patterns generated by the Fuzzy-Apriori-Rare algorithm, *maxsup* = 0.2 as shown in Table 9. In other words, the proposed *FARIM* approach is more suitable for discovering *low-rank* fuzzy rare itemsets, which can be used for detecting learning problems from educational data; see Table 11.

The proposed *FARIM* approach uses rank thresholds to eliminate high-rank patterns. The fuzzy rare itemset (*MS.very-good*, *AS.very-good*) with the support of 4.23%, can only be discovered using the Fuzzy-Apriori-Rare approach. From the fuzzy rare itemset (*MS.very-good*, *AS.very-good*), we can generate a rule "if the attendance score is very good, the mid-term will be very good". Although the support value of this rule is less than the maximum support thresholds $\sigma^R_{\max sup}$, it does not seem meaningful for discovering student learning problems. The reason is that we cannot find those patterns indicating learning problems from this rule.

The above illustration shows that, without the rank threshold, the Fuzzy-Apriori-Rare approach generates many non-meaningful rules, whereas the proposed *FARIM* will not. The proposed *FARIM* approach improves the quality and the accuracy of extracted knowledge. The association rules found using the proposed *FARIM* approach (with the support value $\sigma^R_{\max sup} = 0.2$, rank value $\sigma^R_{\max rnk} = 0.53$ and confidence value $\lambda = 0.95$) are shown in Table 12.

Note that the rules, (*QS.very-low* ⇒ *MS.very-low*) or (*HS.very-low* ⇒ *MS.very-low*) are not generated in Table 12. Therefore, if the student's quiz score or homework score is very low, his/her mid-term score may not be very low. Please refer to the association rules generated by *low-rank* fuzzy rare itemsets in Table 12, found using the *FARIM* approach. Rule #1 for example indicates that if the student's quiz score and homework score are both very low, his/her mid-term score will also be very low. These rules are meaningful for teachers as an indication to provide more assistance. Our findings suggest that lecturers should pay more attention to students who have both low quiz scores and low homework scores.

7. Conclusion

Association rule mining is one of most common data mining techniques for discovering relationships between data. Association rule mining algorithms have a wide variety of applications for various types of data sets. However, they have not so far been used to discover rare itemsets within educational data. This is because previous mining algorithms merely focus on discovering frequent itemsets.

Table 12
Association rules generated by *low-rank* fuzzy rare itemsets.

No.	Rules	Support	Confidence	Rank
1	$(QS.very-low, HS.very-low \Rightarrow MS.very-low)$	7.34%	98.45%	52.59%

MS, mid-term score; FS, final-term score; QS, quiz score; HS, homework score; AS, attendance score.

The main contribution of this study is the development of an effective mining algorithm *FARIM* which can be used to discover *low-rank* fuzzy rare itemsets and then generate meaningful rules from these itemsets. Experimental results show the feasibility of the proposed *FARIM* algorithm for educational data.

There are several issues that remains to be addressed in future research. First, it is assumed here that the membership functions are known in advance. In future, we hope to adopt other data mining technologies, such as clustering [5] or GAs [4], to obtain the membership functions, instead of simply assigning them by experts in the preprocessing phase, as is currently done to avoid an acquisition bottleneck in membership functions. In addition, the values of the four thresholds named minimum support, maximum support, maximum rank and minimum confidence are assigned by experts in this study. In future, however we will attempt to automatically infer these thresholds from the raw data by applying machine learning techniques. Since the proposed *FARIM* approach is based on the Apriori algorithm, it suffers from the same limitations as the Apriori algorithm. Our algorithm generates too many candidate itemsets for large databases when the minsup is set very low. In the future, we attempt to combine the other approaches to refine the proposed *FARIM* approach to discover fuzzy specific rare itemsets more efficiently.

Acknowledgement

This research was supported by the National Science Council of the Republic of China under Contract No. NSC 99-2410-H-166-004.

Appendix A. Proving the correctness of the FARIM algorithm

Before giving the proof, the major steps in the *FARIM* algorithm are briefly summarized. This will enable us to focus on the main ideas of the proof without going into detail:

Input: A database D; membership functions (FS_{f_j}); a predefined minimum support $\sigma^R_{\min sup}$; a predefined maximum support $\sigma^R_{\max sup}$; a predefined maximum rank $\sigma^R_{\max rnk}$; a predefined minimum confidence λ
Output: A set of fuzzy association rules
Method:
// **Phase 1 Call the** *Sup_Transform* Subroutine
// **Phase 2 Call the** *RareItemsets_gen* Subroutine
(1) Calculate the support for each r-item $ic_{j,r}$
(2) Check whether the support for each r-item ic_j is no less than the minimum support, $\sigma^R_{\min sup}$. If it is, assign it to the set of frequent one-itemsets (L_1)
(3) Check whether the support of each r-item ic_j is no larger than the maximum support, $\sigma^R_{\max sup}$. If it is, assign it to the set of *rare* one-itemsets (R_1)
(4) Check whether the rank of each r-itemset in R_1 is no larger than the maximum rank $\sigma^R_{\max rnk}$. If it is, assign it to the set of *low-rank rare* one-itemsets (R^{lr}_1)
(5) Generate candidate set C_{k+1} from L_k
(6) Compute the supports for all r-itemsets in C_{k+1} and then determine R_{k+1} and L_{k+1}
(7) Compute the ranks of all r-itemsets in L_{k+1} and then determine R^{lr}_{k+1}
(8) If L_{k+1} is null, go to phase 3; otherwise, set $k = k + 1$ and repeat steps (3)–(7)
// **Phase 3 Call the** *FAR_gen* Subroutine

The following theorem shows that the patterns found by the algorithm are correct, meaning that every output pattern is *low-rank rare*.

Theorem A.1. *The fuzzy rare itemsets obtained by the FARIM algorithm are correct.*

Proof. Since step 2 sets the minimum support to 0. All items are generated by the algorithm, including fuzzy rare itemsets and fuzzy non-rare itemsets. Step 3 filters out fuzzy non-rare itemsets and leaves fuzzy rare one-itemsets (R_1). Step 4 uses the maximum rank $\sigma^R_{\max rnk}$ to further filter *low-rank* fuzzy rare one-itemsets (R^{lr}_1). Thus, every pattern of length 1 generated by the *FARIM* algorithm is *low-rank* rare. Furthermore, step 5 generates candidate set C_{k+1} from L_k. Since the minimum support is set to 0, all candidate itemsets of length $(k + 1)$ are generated by the algorithm, including fuzzy rare itemsets and fuzzy non-rare itemsets. Steps 6 and 7 filter fuzzy rare itemsets (R_{k+1}) and *low-rank* fuzzy rare itemsets (R^{lr}_{k+1}), respectively. Thus, every fuzzy rare itemset of length $(k + 1)$ generated by the *FARIM* algorithm is *low-rank rare*. □

The following theorem shows that the algorithm is complete, meaning every *low-rank* fuzzy rare itemsets will be found by the algorithm.

Theorem A.2. *The FARIM algorithm can find every low-rank fuzzy rare itemset.*

Proof. Let $B = \langle ic_1, ic_2, \ldots, ic_r \rangle$ denote a *low-rank* fuzzy rare itemset with k r-items, where ic_i is a r-itemset for $1 \leqslant i \leqslant r \leqslant k$. If $k = 1$, the pattern degenerates into an r-item. Obviously, step 4 can find pattern B because it finds all *low-rank* fuzzy rare r-items. Assume that the algorithm can find all *low-rank* fuzzy rare itemsets of k r-items. Let $B' = B + ic_{k+1}$ be a *low-rank* fuzzy rare itemset of $k + 1$ r-items. We need to show that the algorithm can find B'. This can be verified as below:

(1) Since step 2 sets the minimum support to 0, step 5 will generate candidate set C_{k+1}, including, including all *low-rank* fuzzy rare itemsets, from L_k.
(2) Step 6 generates fuzzy rare itemsets (R_{k+1}) by using the max-support threshold.
(3) Step 7 generates *low-rank* fuzzy rare itemsets (R^{lr}_{k+1}), including ic_{k+1}, by using the max-rank threshold.

This ensures that, after extending one r-item, B' is still correct. Therefore, the *FARIM* algorithm can find every *low-rank* fuzzy rare itemsets. □

References

[1] M. Adda, L. Wu, Y. Feng, Rare itemset mining, in: Sixth International Conference on Machine Learning and Applications, 2007, pp. 73–80.
[2] R. Agrawal, T. Imielinski, A. Swami, Mining association rules between sets of items in large databases, in: Proceedings of ACM SIGMOD, 1993, pp. 207–216.
[3] R. Agrawal, R. Srikant, Fast algorithms for mining association rules, in: Proceedings of the 20th International Conference on Very Large Data Bases, 1994, pp. 487–499.
[4] A. Arslan, M. Kaya, Determination of fuzzy logic membership functions using genetic algorithms, Fuzzy Sets and Systems 118 (2) (2001) 297–306.
[5] M. Berry, G. Linoff, Data Mining Techniques: For Marketing, Sales, and Customer Support, Wiley, NY, 1997.
[6] C.M. Chen, Y.L. Hsieh, S.H. Hsu, Mining learner profile utilizing association rule for web-based learning diagnosis, Expert Systems with Applications 33 (1) (2007) 6–22.

[7] Y.L. Chen, C.H. Weng, Mining association rules from imprecise ordinal data, Fuzzy Sets and Systems 159 (4) (2008) 460–474.
[8] Y.L. Chen, C.H. Weng, Mining fuzzy association rules from questionnaire data, Knowledge-Based Systems 22 (1) (2009) 46–56.
[9] M. Delgado, N. Marin, D. Sanchez, M.A. Vila, Fuzzy association rules: general model and applications, IEEE Transactions on Fuzzy Systems 11 (2) (2003) 214–225.
[10] J. Han, W. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufman, San Francisco, CA, 2001.
[11] T.P. Hong, C.S. Kuo, S.C. Chi, Mining association rules from quantitative data, Intelligent Data Analysis 3 (5) (1999) 363–376.
[12] T.P. Hong, C.S. Kuo, S.L. Wang, A fuzzy AprioriTid mining algorithm with reduced computational time, Applied Soft Computing 5 (1) (2004) 1–10.
[13] T.P. Hong, K.Y. Lin, S.L. Wang, Fuzzy data mining for interesting generalized association rules, Fuzzy Sets and Systems 138 (2) (2003) 255–269.
[14] M.H. Hsu, A personalized English learning recommender system for ESL students, Expert Systems with Applications 34 (1) (2008) 683–688.
[15] Y.C. Hu, G.H. Tzeng, Elicitation of classification rules by fuzzy data mining, Engineering Applications of Artificial Intelligence 16 (7–8) (2003) 709–716.
[16] Y.C. Hu, R.S. Chen, G.H. Tzeng, Discovering fuzzy association rules using fuzzy partition methods, Knowledge-Based Systems 16 (3) (2003) 137–147.
[17] G.J. Hwang, C.L. Hsiao, C.R. Tseng, A computer-assisted approach to diagnosing student learning problems in science courses, Journal of Information Science and Engineering 19 (2003) 229–248.
[18] W. Lian, D.W. Cheung, S.M. Yiu, An efficient algorithm for finding dense regions for mining quantitative association rules, Computers and Mathematics with Applications 50 (3–4) (2005) 471–490.
[19] H.L. Lim, C.S. Lee, Processing online analytics with classification and association rule mining, Knowledge-Based Systems 23 (3) (2010) 248–255.
[20] P. Markellou, I. Mousourouli, S. Spiros, A. Tsakalidis, Using semantic web mining technologies for personalized e-learning experiences, in: Proceedings of the Web-Based Education, 2005, pp. 461–826.
[21] G. Masuda, N. Sakamoto, A framework for dynamic evidence based medicine using data mining, in: Proceedings of the 15th IEEE Symposium on Computer-Based Medical Systems, 2002, pp. 117–122.
[22] A. Merceron, K. Yacef, Mining student data captured from a web-based tutoring tool: initial exploration and results, Journal of Interactive Learning Research 15 (4) (2004) 319–346.
[23] R.J. Miller, Y. Yang, Association rules over interval data, in: Proceedings of the 1997 ACM SIGMOD International Conference on Management of data SIGMOD, vol. 26, Issue 2, 1997, pp. 452–461.
[24] B. Minaei-Bidgoli, P. Tan, W. Punch, Mining interesting contrast rules for a web-based educational system, in: International Conference on Machine Learning Applications, 2004, pp. 1–8.
[25] C. Romero, S. Ventura, Educational data mining: a survey from 1995 to 2005, Expert Systems with Applications 33 (1) (2007) 135–146.
[26] C. Romero, S. Ventura, E. García, Data mining in course management systems: Moodle case study and tutorial, Computers & Education 51 (1) (2008) 368–384.
[27] R. Srikant, Q. Vu, R. Agrawal, Mining association rules with item constraints, in: SIGMOD International Conference on Management of Data, 1996, pp. 1–12.
[28] L. Szathmary, A. Napoli, P. Valtchev, Towards rare itemset mining, in: 19th IEEE International Conference on Tools with Artificial Intelligence, 2007, pp. 205–312.
[29] P.N. Tan, M. Steinbach, V. Kumar, Introduction to Data Mining, Pearson Addison-Wesley, 2006.
[30] S.S. Tseng, P.C. Sue, J.M. Su, J.F. Weng, W.N. Tsai, A new approach for constructing the concept map, Computers & Education 49 (3) (2007) 691–707.
[31] P. Yu, C. Own, L. Lin, On learning behavior analysis of web based interactive environment, in: Proceedings of the Implementing Curricular Change in Engineering Education, 2001, pp. 1–10.