International Journal of Computational Intelligence and Informatics, Vol. 1 : No. 3, October - December 2011

A Modified Algorithm for Generating Single Dimensional Fuzzy Itemset Mining

D. Ashok Kumar
Department of Computer Science
Government Arts College
Trichy - 620 022, India
akudaiyar@yahoo.com

R. Prabamanieswari
Department of Computer Science
Govindammal Aditanar College for Women
Tiruchendur - 628 215, India
prabacs_2006@yahoo.com

Abstract

Mining frequent itemsets from transaction database is a fundamental task for Association Rules. Apriori algorithm is an influential algorithm for mining frequent itemsets using Boolean values. There are different motivations for a fuzzy approach to Association Rule Mining. An Algorithm for Generating Single Fuzzy Association Rule Mining is based on human intuitive such as the larger number of items purchased in a transaction means that the degree of association among the items in the transaction may be lowered. The proposed approach modifies the above said Fuzzy Association Rule Mining algorithm and compares Apriori and the mentioned Fuzzy Association Rule Mining algorithm. The proposed approach calculates the support value based on fuzzy t-norm namely intersection and finds the subsets of a frequent itemset partially. Therefore, it reduces the complexion of finding each subset of a frequent itemset.

Keywords: frequent itemset, fuzzy set, fuzzy intersection

1. Introduction

Data mining is a form of knowledge discovery essential for solving problems in a specific domain. The rapid growth of interest in data mining is due to the i) falling cost of large storage devices and increasing ease of collecting data over networks; ii) development of robust and efficient machine learning algorithms to process this data; and iii) falling cost of computational power, enabling use of computationally intensive methods for data analysis.

Since the introduction in 1993 by Agrawal [1], the frequent itemset and association rule mining problems have received a great deal of attention. The hundreds of research papers have been published presenting new algorithms or improvements on existing algorithms to solve these mining problems more efficiently. But, Apriori [1] is an influential algorithm for mining frequent itemsets for Boolean Association Rules. Generally, based on this algorithm, support of an itemset is determined by just counting the number of occurrences of the itemset in every record of transaction (shopping cart), without any consideration to human intuitive. Recently, the fuzzy set theory [3] has been used more and more frequently in intelligent systems because of its simplicity and similarity to human reasoning. Based on the concept, such as the larger number of items purchased in a transaction means that the degree of association among the items in the transaction may be lowered, is discussed well by utilizing fuzzy sets in the market basket analysis[2]. Generate Frequent Fuzzy Pattern (GFFP) algorithm discussed in [4] for generating fuzzy association rules from frequent fuzzy itemsets is similar to generating Boolean Association Rules. Pratima Gautam et al [5] defined the fuzzy set similar to Rolly Intan [2] but derived the multiple-level association rules under different supports. In [6], an algorithm for generating fuzzy association rules with multidimensional attributes utilizing fuzzy set based on human intuitive is discussed. It is similar to [2]. Therefore, this paper also follows the human intuition and uses the same membership functions defined in [2], but it considers all transactions instead of considering qualified transactions based on δ- maximum threshold. It also uses t-norm in fuzzy set such as $T_M(x, y) = \min(x, y)$ for finding the support value.

Section 2 illustrates the importance of frequent itemset mining and Apriori algorithm. Section 3 describes the existing algorithm [2]. Section 4 gives the proposed algorithm for determining frequent itemsets. Section 5 demonstrates the algorithm with a worked example. Section 6 discusses the experimental results. Finally conclusion is given in Section 7.

2. Importance of Frequent itemset Mining and Apriori

Frequent patterns are itemsets, subsequences or substructure that appear in a data set with frequency no less than a user-specified threshold. Itemsets are collections of items that co-occur in data. Mining frequent itemsets from transaction database is a fundamental task for several forms of knowledge discovery such as association rules, sequential patterns and classification. Historically their primary use in data mining has been as an intermediate step in discovery of association

rules [1]. The original motivation for searching association rules came from the need to analyze so called supermarket transaction data, that is, to examine customer behaviour in terms of the purchased products.

The name of Apriori is based on the fact that the algorithm uses prior knowledge of frequent itemset properties. Apriori employs an iterative approach known as a *level-wise* search, Where *k*-itemsets are used to explore (*k+1*)-itemsets. First, the set of frequent 1- itemsets is found, denoted by L_1. L_1 is used to find L_2, the set of frequent 2-itemsets, which is used to find L_3, and so on, until no more frequent k-itemsets can be found. Fig. 1 shows the Apriori-Itemset Mining algorithm.

Algorithm **Apriori – Itemset Mining**
Input: D, σ
Output: $F(D, \sigma)$

$C_1 := \{\{i\} \mid i \in I\}$
$k := 1$
While $C_k \neq \{\}$ *do*
// Compute the supports of all candidate itemsets
for all transactions $(tid, I) \in D$ *do*
for all candidate itemsets $X \in C_k$ *do*
if $X \subseteq I$ *then*
$X.Support++$
end if
end for
end for
// Extract all frequent itemsets
$F_k := \{X \mid X.\text{sup}\,port \geq \sigma\}$
// Generate new candidate itemsets
for all $X, Y \in F_k$, $X[i] = Y[i]$ *for* $1 \leq i \leq k-1$, *and*
$X[k] < Y[k]$ *do*
$I = X \bigcup \{Y[k]\}$
if $\forall J \subset I, |J| = k : J \in F_k$ then
$C_{K+1} := C_{K+1} \cup I$
end if
end for
$k++$
End while

Fig. 1 Apriori-Itemset Mining algorithm

3. An Algorithm for Generating Single Dimensional Fuzzy Association Rule Mining

Apriori algorithm ignored the number of items in a shopping cart in determining relationship of the items. In order to find the relationship among the items, this algorithm considered number of items in each record of transaction. It suggested that the increasing number of items will reduce the relationship among the items (each items has a lower fuzzy membership value in a transaction). It used maximum item threshold δ to determine maximum number of items in a transaction for selecting qualified transaction i.e.) the number of items in its transaction is not greater than δ. It also considered the minimum support for k-itemsets, denoted by $\beta_k \in (0, |M|)$ where $|M|$) is the number of qualified transaction. The following membership functions are used in this algorithm.

A fuzzy membership function μ is a mapping:
$\mu_1 k : M \rightarrow [0,1]$ as defined by:

$$\mu_1 k(T) = \inf_{i \in I^k} \left\{ \frac{\eta_T(i)}{card(T)} \right\}, \forall T \in M \qquad (1)$$

where $I^k \subseteq \Im; T$ be a qualified transaction in which T can be regarded also as a subset of items $(T \subseteq \Im)$;
A Boolean membership function η is a mapping:
$\eta_T : \Im \rightarrow \{0,1\}$ as defined by:

$$\eta_T(i) = \begin{cases} 1, & i \in T \\ 0, & otherwise \end{cases} \qquad (2)$$

such that if an item, i, is an element of T then $\eta_T(i) = 1$,
otherwise $\eta_T(i) = 0$.
The support for every candidate k-itemset I^k is calculated using the equation

$$\text{Support}\left(I^k\right) = \sum_{T \in M} \mu_1 k(T) \qquad (3)$$

and the candidate k-itemset I^k can be considered as the frequent itemset L_k if and only if support $\left(I^k\right) \geq \beta_k$. The largest frequent itemset L_k is calculated based on δ *i.e.)* K$\leq \delta$.

4. Proposed Algorithm

The proposed algorithm uses the same membership functions defined in [2], but it considers all transactions instead of considering qualified transactions based on δ (maximum item threshold). It uses fuzzy t-norm T_M (x, y) = min (x, y) for calculating support value i.e.) it increments the support value based on fuzzy intersection value>0.0. When finding

intersection among the items, if any one item has a value 0, then it does not continue to find the intersection for the remaining items. Therefore, it reduces the time. And also, it does not find out all the subsets for finding a frequent itemset. It checks only the subsets contained in the generated candidate other than the items used for generating candidates and prune the generated candidate based on the absent of the specified subset and selects the candidate for finding a frequent candidate i.e. it finds only the reduced number of subsets. But Apriori and the mentioned Fuzzy Association Rule Mining algorithm [2] find all the subsets of a frequent itemset. Hence, the proposed approach reduces the time comparing to both algorithms.

The proposed algorithm is given in Fig. 2:

Algorithm

Input: *Fuzzy Database, support factor*

Output: *frequent fuzzy itemsets*

Method:

L_1 = {frequent 1-itemsets where supp $\geq$ support factor}

number of items in a combination (c1) = 1

k:=2

while (L_{k-1} != φ)

{

call_generate_new (c1)

c1++

k++

}

Procedure call_generate_new(c1)

generate candidates (L_{k-1})

for each generated candidates c$\in C_k$

{

if (c1=1) frequent()

else

{

get all partial subset(c)

if (all partial subset(c) $\in$ L_{k-1})

frequent()

}

}

Procedure frequent()

for each transaction T$\in$ D

{

find fuzzy intersection(c) // during fuzzy intersection, if result is zero, then skip that Transaction

if (fuzzy intersection(c) > 0.0)

c.supp++

L_k = {c | c.supp $\geq$ support factor}

}

Fig. 2 Proposed Algorithm

5. Worked Example

In general, a transactional database consists of a file in which each record represents a list of items purchased in a transaction. Simply, a transaction includes a unique transaction identity number (*trans_id*) and the list of items making up the transaction. A transactional database may have additional information regarding the sale such as customer ID, date of transaction, etc.

Table 1 shows a sample transactional database.

Table 1: A Transactional Database

Trans_id	***List of items***
T1	i1, i2, i3, i4
T2	i2, i3, i4, i5
T3	i1, i2, i4
T4	i3, i4, i5
T5	i1, i4

A worked example is given to understand well the concept of the proposed algorithm and how the process of generating fuzzy frequent itemset is performed step by step. The process is started from a given transactional database as shown in Table 1. Here, the support factor is considered as 2. The fuzzy set is constructed based on Rolly Intan's same membership function.

Table 2: Fuzzy Set

Item / Trans_Id	i1	i2	i3	i4	i5
T1	0.25	0.25	0.25	0.25	0.00
T2	0.00	0.25	0.25	0.25	0.25
T3	0.33	0.33	0.00	0.33	0.00
T4	0.00	0.00	0.33	0.33	0.33
T5	0.5	0.00	0.00	0.5	0.00

Here, each item is replaced by $\frac{1}{total}$ number of items in a transaction.

The support count of each 1-itemset is calculated as counting each 1- itemset based on its fuzzy value greater than 0.0.

{i1} = {0.25/T1,0.33/T3,0.5/T5} supp = 3;

{i2} = {0.25/T1,0.25/T2,0.33/T3} supp = 3;

{i3} = {0.25/T1,0.25/T2,0.33/T4} supp = 3;

{i4} = {0.25/T1,0.25/T2,0.33/T3,0.33/T4,0.5/T5} supp = 5;

{i5} = {0.25/T2,0.33/T4} supp = 2;

All 1-itemsets's support count ≥ 2.

Therefore, L_1 = {i1, i2, i3, i4, i5}.

Iteration -1

c1 = 1;

Generating 2-itemsets, finding fuzzy intersection value > 0.0 and finding their support ≥ 2 are:

{i1, i2} = {0.25/T1, 0.33/T3} supp = 2;

{i1, i4} = {0.25/T1, 0.33/T3} supp = 2;

{i2, i3} = {0.25/T1, 0.25/T2} supp = 2;

{i2, i4} = {0.25/T1, 0.25/T2,0.33/T3) supp = 3;

{i3, i4} = {0.25/T1, 0.25/T2, 0.33/T4) supp = 3;

{i3, i5} = {0.25/T2, 0.33/T4} supp = 2;

{i4, i5} = {0.25/T2, 0.33/T4} supp = 2;

Here, there is no need for checking subsets of each 2-itemset because all the generated candidates come from the existing frequent 1-itemset list. Therefore, L_2 = {i1, i2; i1, i4; i2, i3; i2, i4; i3 i4; i3 i5; i4 i5}.

Iteration -2

c1 = 2;

Generating 3-itemsets, finding fuzzy intersection value > 0.0 and finding their support ≥ 2 are:{i1, i2, i4} = {0.25/ T1, 0.33/T3} supp = 2;

{i2, i3, i4} = {0.25/ T1, 0.25/T2} supp = 2;

{i3, i4, i5} = {0.25/ T2, 0.33/T4) supp = 2;

Here, for finding {i1,i2,i4}, it is not necessary to find all the subsets such as {i1,i2},{i1,i4}and {i2,i4}, but it is only necessary to find {i2,i4} because {i1,i2} and {i1,i4}are already in L_2. Similarly, the subset {i3,i4} is only checked for finding {i2,i3,i4} and the subset {i4,i5} is only checked for finding {i3,i4,i5}.

Therefore, L_3 = {i1, i2, i4; i2, i3, i4; i3, i4, i5}

Iteration -3

c1 = 3;

Generating 4-itemsets, finding fuzzy intersection and finding their support ≥ 2 are:

L_4 = {}

Therefore, it does not continue further Iteration.

In this approach, when finding fuzzy intersection (i.e. for finding minimum value among the items), if any one item has a value 0, then the intersection is not continued for the remaining items in a transaction. It reduces the time for finding frequent itemset.

6. Experimental Results

Apriori, Rolly Intan's algorithm and the proposed algorithm are experimented with the mushroom dataset. The mushroom dataset contains the characteristics of various species of mushrooms. It has 119 items and 8124 transactions. It can be obtained from the UCI repository of machine learning databases. The minimum, maximum and average length of each transaction is 23.

The algorithms are implemented using C# language and carried on the computer with the configuration such as Intel(R) Core(TM) i3CPU, 3 GB RAM, 2.53 GHz Speed and Windows 7 Operating System.

The following fig. 3 shows the performance of three algorithms.

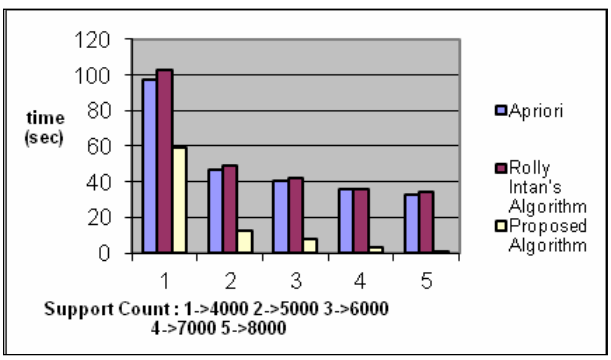

Fig. 3 Performance of three Algorithms

It clearly shows the proposed algorithm takes lesser time comparing to other two algorithms.

7. Conclusion

This paper modifies the existing algorithm for determining frequent itemset using fuzzy sets. It defines the fuzzy sets based on the existing algorithm and applies the fuzzy set operation (intersection) on the selected items (based on finding the specified subset) in a transaction. Finally it finds the frequent itemset based on the minimum support factor. It also illustrates the reduction time in finding subsets in both algorithms.

References

[1] Agrawal. R., Imielimski. T and Swami. A.N., "Mining Association Rules between sets of items in large database", Proceedings of ACM SIGMOD International Conference Management of Data, ACM Press, pp. 207-216, 1993.

[2] Rolly Intan, "An Algorithm for Generating Single Dimensional Fuzzy Association Rule Mining", Journal Informatika Vol. 7, No. 1, pp. 61-66, MEI 2006.

[3] Zadeh L.A., "Fuzzy Sets and systems", International Journal of General Systems, Vol. 17, pp. 129-138, 1990.

[4] Jitao Zhao and Lin Yao, "A General Framework for Fuzzy Data Mining", International Conference on Computational Intelligence and Software Engineering (CISE), pp. 1-3, 2010.

[5] Pratima Gautam et al , "A model for mining multilevel fuzzy association rule in database", Journal of Computing, Volume 2, Issue 1, pp. 2151-9617, January 2010.

[6] Neelu Khare et al, "An Algorithm for Mining Multidimensional Fuzzy Association Rules", International Journal of Computer Science and Information Security, vol. 5, No. 1, pp. 72-76, 2009.

D. ASHOK KUMAR did his Master degree in Mathematics and Computer Applications in 1995 and completed Ph.D., on Intelligent Partitional Clustering Algorithm's in 2008, from Gandhigram Rural Institute-Deemed University, Gandhigram, Tamilnadu, India. He is currently working as Senior Grade Assistant Professor and Head in the Department of Computer Science, Government Arts College, Trichy, Tamilnadu, India. His research interest includes Pattern Recognition and Data Mining by various soft computing approaches viz., Neural Networks, Genetic Algorithms, Fuzzy Logic, rough set, etc.

R. PRABAMANIESWARI received the Master's degree in Computer Applications in 1993 and Master of Philosophy in Computer Science in 2002 from Manonmaniam Sundaranar University, Tirunelveli, Tamilnadu, India. She is currently working towards her Ph.D in Manonmaniam Sundaranar University. Her current research interests include the areas of Data Mining and Database Systems.