Without Expert Fuzzy based Data Mining based on Fuzzy Similarity to Mine New Association Rules

Gagan Dhawan
Assistant Professor
Department Of Computer Science & Engineering,
R. P. Inderprasth Institute of Technology

Aakanksha Mahajan
M.Tech Research Scholar
Department of Computer Science & Engineering, Doon Valley Institute of Engineering And Technology

ABSTRACT
The problem of mining association rules in a database are introduced. Most of association rule mining approaches aim to mine association rules considering exact matches between items in transactions. A new algorithm called "Without expert fuzzy based data mining Based on Fuzzy Similarity to mine new Association Rules " which considers not only exact matches between items, but also the fuzzy similarity between them. In this paper their should not be a requirement to have an expert for finding similarity between items. Without expert fuzzy based(WEFB) Data Mining Based on fuzzy Similarity to mine new Association Rules uses the concepts of without expert to represent the similarity degree between items, and proposes a new way of obtaining support and confidence for the association rules containing these items. The problem is to find all association rules that satisfy user-specified minimum support and minimum confidence constraints. This paper then results that new rules bring more information about the database.

Keywords
Data mining, Association Rules, Support, Confidence, Fuzzy logic

1. INTRODUCTION TO DATA MINING
"Data Mining (DM)" is a technique to extract nontrivial regularities or relationships in databases. "Knowledge Discovery in Databases (KDD)" has almost the same meaning. The amount of data stored in information systems is rapidly increasing due to the advance of computer technologies. However, analysis of these data still requires human experts' skills of statistics. Since it is practically impossible to assign a statistician to each data analysis task, it happens that huge amount of data is tend to be kept unused. The purpose of data mining [1] [3] is to extract nontrivial regularities or relationships as a piece of knowledge in databases. This technique can provide users with a very powerful tool for exploiting vast amount of stored data. A rule is one of the typical knowledge representations. For example, an interesting pattern between data fields is expressed as

"if xl is al and x1 is a1 then y is b

with probability **p."** However, it is difficult to apply conventional rule induction algorithms to data mining, because of the following characteristics of real world data.

Data volume: Millions of records can be found in many databases. To process such large amount of data, a very efficient search algorithm is required.

Noise and uncertainty: Data values may contain certain level of noise by statistical fluctuations or human errors. To deal with this, the algorithm should be able to extract probabilistic rules.

Incompleteness: Most databases are not designed for data mining. Therefore, we cannot expect all the necessary fields are prepared in the target database. Utilization of domain knowledge and one or user-interactive analysis environment becomes necessary. In this exhibition, a data mining system is introduced. The main feature of this system is a specially designed rule induction algorithm which extracts useful pattems in databases.

2. IMPROVED DATA MINING BASED ON FUZZY WEIGHTED ASSOCIATION RULES
Data Mining has been researched a lot due to its utility in many applications, and one of its most used tasks is Association Rule Mining. Given a set of transactions, where each transaction is a set of items, an association rule is an expression X => Y, where X and Y are sets of items (or item sets). The meaning of such a rule is that transactions which contain items in X tend to also contain items in Y. The support of the rule X => Y is the percentage of transactions that contain both X and Y. The confidence of the rule X => Y is the percentage of transactions containing X that also contain Y an example of an association rule is "90% of transactions that contain bread also contain butter; 3% of all transactions contain both of these items."The 90% is referred to as confidence and the 3%, the support of the rule. The problem of mining association rules is to find rules having minimum support and confidence. Many algorithms were developed to solve the problem of mining association rules. In general, new approaches were motivated by finding new ways of dealing with different attributes types or increasing computational performance. However, new approaches could address other issues. In this paper, we concern about fuzzy similarities to mined data and their should not be a requirement to have an expert for finding similarity between items.

Known algorithms only consider exact matches when mining frequent item sets, not generating some association rules which could bring important information.

In our approach, besides exact matches, the fuzzy similarity between items is also taken on account. For example, consider the set of transactions shown in Table 1.

TID	attribute1	attribute2
1	Chair	Table
2	Sofa	Desk
3	Chair	Desk
4	Chair	Table

Table1. A set of transaction examples

If this set of transactions were mined by a traditional association rule mining algorithm, the following association rules would be obtained:
Chair => table (support 50%, confidence 67%)
Sofa => desk (support 25%, confidence 100%)
Chair => desk (support 25%, confidence 33%)
Thus, if a minimum support of 50% and a minimum confidence of 60% were established, the only rule generated would be chair=> table. In this situation, only strings of characters are being

considered, and as they have the same characters, with the same order and the same length, the mining algorithm will recognize a match. Table and desk, for example, are totally different words, but it does not mean they are totally different items. If we semantically analyze the words table and desk, we can consider them similar (both are furniture and have similar utilities, for example). In this case, there is not an exact match, but there is a kind of "fuzzy similarity match", which can be also useful to find relevant association rules and therefore important information. That is what traditional approaches can not reveal: association rules including semantically similar items. To make it possible, in this paper we present an algorithm called WEFB data mining.

3. ALGORITHM

3.1 Fuzzy Similarity

"Fuzzy Logic is basically a multi-valued logic that allows intermediate values to be defined between conventional evaluations like yes/no, true/false, black/white, etc. Fuzzy Logic[6] was first invented as a representation scheme and calculus for uncertain or vague notions. This fuzzy similarity between items is ignored by traditional algorithms, what can make them lose important information. In this paper, we present a new algorithm called WEFB data mining. In WEFB data mining, the Fuzzy similarities is basically used to mined data and their should not be a requirement to have an expert for finding similarity between items. In this section we show how WEFB data mining detects these fuzzy similarity associations and uses them to get important association rules. For example in figure 1, we take several furniture items like sofa, chair and seat where one can sit, board, desk and table where one can place things on them, and cabinet, cupboard and wardrobe where one can store things. We apply fuzzy rules on these item sets to generates more association rules likewise sofa is basically used for seating, in addition of these we also check sofa is used for placing or storing things or not. So we apply fuzzy rules on sofa. we started testing on one of item sofa having following attributes seating(very-2 low, very low, low, medium, high, very high,very-2 high), placing((very-2 low, very low, low, medium, high, very high,very-2 high),storing((very-2 low, very low, low, medium, high, very high,very-2 high). On the basis of these attributes values we find there is no need to have an expert which is used to represent similarity degrees. Thus the need of an expert is vanished and the results can be obtained even by a layman. Thus the data mined this way generate more association rules i.e. more information about data which can be useful for decision making. An example given for furniture store for the fuzzy similarity degree calculation is

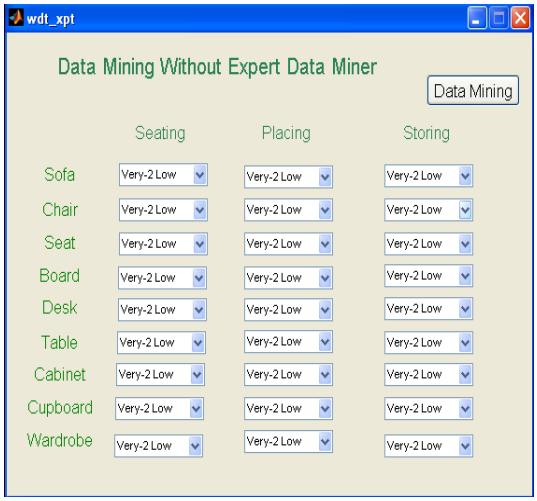

Figure1.

3.2 Algorithm Structure

WEFB data mining is based on Apriori and, as an association rule mining algorithm, it needs user-provided minimum support and minimum confidence parameters to run. Moreover, by using fuzzy logic concepts[7], WEFB data mining also needs a user provided parameter which indicates the minimum similarity degree desired, called minsim. Thus, there are the following parameters:

• minsup, which indicates the minimum support;
• minconf, which represents the mininum confidence;
• minsim, which is the minimum similarity degree necessary to consider two items similar enough, and then associate them during mining.

All of these parameters are expressed by a real value in the interval [0, 1]. The steps performed by WEFB data mining are shown below

1. Data Scanning: Identifying items and their domains
2. Determining similarity degrees between items for each domain
3. Identifying similar items
4. Generating candidates
5. Calculating the weight of candidates
6. Evaluating candidates
7. Generating rules

Now, consider as an example a table containing transactions of buys from a furniture store (Table 2), where Tid is an identifier for each transaction, whereas Dom1, Dom2 and Dom3 contain items bought by the furniture store customers.

Moreover, suppose henceforth that we have the following parameter values:

• minimum support (minsup) = 0.45
• minimum confidence (minconf) = 0.3
• minimum similarity (minsim) = 0.8

Tid	Dom1	Dom2	Dom3
10	Chair	Table	wardrobe
20	Sofa	Desk	cupboard
30	Seat	Table	wardrobe
40	Sofa	Desk	cupboard
50	Chair	Board	wardrobe
60	Chair	Board	cupboard
70	Chair	Desk	cupboard
80	Seat	Board	cabinet
90	Chair	Desk	Cabinet
100	Sofa	Desk	cupboard

Table2. Transactions of buys from a furniture store

3.2.1 Data Scanning

The first step is a data scanning that identifies items in the database. WEFB data mining identifies each item, generating 1-itemsets (item sets with size one). Moreover, in this step each item is associated to a domain, which is important because they make possible to relate items according to their similarity only when is convenient — that is, if they belong to the same domain. When mining relational tables, domains can be defined by the column where the item is. Thus, considering the furniture store example, after data scanning we have items and domains identified, as shown in Table 3.

Items	Domain
sofa, chair, seat	Dom1
board, desk, table	Dom2
cabinet, cupboard, wardrobe	Dom3

Table3. Items and domains identified by data scanning

In this example, domain Dom1 contains items of furniture where one can sit, domain Dom2 contains items of furniture where one can place things on them, and domain Dom3 contains items of furniture where one can store things. Each domain contains items used in similar situations, what makes domains identification semantically coherent. The number of items belonging to domain determines its size. Thus, all domains in Table 3 have size 3.

3.2.2 Determining Similarity Degrees

After having items and their domains identified, it is time to determine the values of similarity relations within each domain. These values must be supplied by a domain specialist (usually the user himself). This task corresponds to one of the steps of KDD [4], prior to the step of data mining. Alternatively, it would be possible to obtain these values automatically, through a rule or method. However, to determine the similarity values between items so that the semantics is considered, it is necessary to adopt a way of reproducing, with high fidelity, the capacity of the human mind of doing this. Any rule chosen to determine these values automatically will consider non-semantic factors, decreasing the quality of the analysis realized and this way going against the objective of the fuzzy similarity data mining, which is to enrich the analysis and consequently enrich the information obtained from the rules[8]. In each domain, the similarity degree values are stored in a similarity matrix. In the furniture store example, 3 domains were identified, and the correspondent similarity matrices can be seen in Table 4. The values in the matrices inform the similarity degree between the items of the domain. For example, chair is 70% similar to sofa. Next subsection shows how each similarity matrix is consulted to identify similar items.

3.2.3 Generating Candidates

The way candidates are generated is very similar to the way it is done in Apriori. However, in WEFB data mining, besides items identified during the data scanning step, fuzzy items — which represent fuzzy associations obtained in the step of identifying similar items — also integrate the generated candidates. At the end of this step, we have the set of *k*-item set candidates, which is submitted to the step of calculating the weight of candidates.

3.2.4 Calculating the Weight of Candidates

In this step, the weight of each item set candidate is calculated. The weight of an item set corresponds to the number of its occurrences in the database. In WEFB data mining, differently from what happens in A priori, an item set can have fuzzy items, hence called fuzzy item set. The notation item1~item2, has the following meaning: if item1 and item2 are very similar, they can be considered as being practically identical; thus, if occurrences of item1 or item2 are found in the database, they will be associated and, together with the similarity degree between items, they will compose a fuzzy occurrence of item1~item2. Therefore, we need to know if the item set is fuzzy or not, before calculating its weight: if the item set is not fuzzy, we calculate its weight in the conventional way, counting its exact occurrences; if the item set is fuzzy, we shall consider its fuzzy occurrences to obtain its weight. To understand how fuzzy occurrences happen, suppose that the similarity degree between item1 and item2 is 0.8. In this case, each occurrence of item2 in the database can be considered equal to 80% of item1 occurrence. Consequently, for each item1 occurrence we sum one item1 occurrence (of course), and for each item2 occurrence we sum 0.8 item1 occurrence (Table 4– situation A).

Tid	Dom1	
10	item1	1.0
20	item1	1.0
30	item2	0.8

Situation A

Tid	Dom1	
10	item1	0.8
20	item1	0.8
30	item2	1.0

Situation B

Table4. Fuzzy Occurrences

The problem can also be seen in the contrary manner, summing one item2 occurrence for each item2 occurrence and 0.8 for each item1 occurrence (Table 4– situation B). Notice that, for situation A, the fuzzy occurrences totalize the value of 2.8 (1.0 + 1.0 + 0.8), whereas for situation B fuzzy occurrences totalize the value of 2.6 (0.8 + 0.8 + 1.0). Hence, depending on situation, the result obtained for the same similar items could be different. To avoid this distortion, it is necessary to balance this counting. To do that, consider weight(item1) as the number of item1 occurrences, weight(item2) as the number of item2 occurrences; and sim(item1,item2) as the similarity degree between item1 and item2. Thus, for situation A in Table 4, the number of occurrences is given by the expression.

$$\text{weight}(\text{item}_1) + \text{weight}(\text{item}_2) \times \text{sim}(\text{item}_1, \text{item}_2)$$

In the same way, for situation B in Table 7, the number of occurrences is given by the expression.

$$\text{weight}(\text{item}_1) \times \text{sim}(\text{item}_1, \text{item}_2) + \text{weight}(\text{item}_2)$$

We adopt the arithmetic average between situations A and B to balance the two situations, getting the fuzzy weight of item1~item2 through the Equation 1.

$$\text{Fuzzy Weight} = \frac{[\text{weight}(\text{item1}) + \text{weight}(\text{item2})][1 + \text{sim1}(\text{item1}, \text{item2})]}{2}$$

Equation1. Fuzzy weight for two similar items

Equation 1 is useful to calculate the weight of fuzzy items formed by an association of only two similar items. After this, itemset candidates are evaluated in the next step of WEFB data mining.

3.2.5 Generating Rules

Association rules have antecedents (items left of arrow) and consequents (items right of arrow), as shown in Figure 2.

Antecedent →Consequent

Figure2. Antecedent and consequent of the rule

The support corresponds to the weight divided by the number of rows (or total of transactions) in the database (Equation 2).

$$\text{Support} = \frac{\text{weight(itemsets)}}{\text{number of rows in the database}}$$

Equation2. Support of the item set

The confidence, given by Equation 3,

$$\text{Confidence} = \frac{\text{Support(rule)}}{\text{Support(anteecedent)}}$$

Equation3. Rule confidence

When WEFB data mining is concluded, all valid rules are exhibited, showing antecedent, consequent, support and confidence of each rule, in the format shown in Figure 3.

Antecedent→ Consequent sup = < support value > conf = <confident value>

Figure3. Association Rule Format

In WEFB data mining, antecedents and consequents of the rule can contain fuzzy items, and the values of support and confidence reflect the influence of the similarity degree between items in their calculations.

4. TESTS

We realized some tests to compare the results obtained with WEFB data mining and Apriori, using real data about furniture store. We started testing our first set of data, named FURNITURE STORE, containing transactions with the following attributes. These similarity values are manually decided.

We mined FURNITURE STORE using Apriori with parameters minsup = 40 and minconf = 40, obtaining the rules shown in Figure 8. We also mined FURNITURE STORE using WEFB data mining with the parameters minsup = 40, minconf = 40 and minsim = 80, obtaining the rules shown in Table 5.

Test with Apriori over the set FURNITURE STORE, with minsup = 40 and minconf = 40,Itemsets pair above minimum support and minimum confidence rule:
Rules generated Chair → Sofa sup= 50% conf= 66.6% Sofa → Chair sup= 50% conf= 66.6%

Table5. Test With Apriori over the Set FURNITURE STORE

In Table 6, the underlined rules are those ones which are obtained by WEFB data mining, but are not obtained by Apriori. The additional rules bring more information, which can be useful for decision making. When the association rule contains fuzzy items, its support and confidence values are

Test with WEFB data mining over the set FURNITURE STORE, with minsup = 40 and minconf = 40,Itemsets pair above minimum support and minimum confidence rule:
Rules generated Chair~sofa → table sup= 50% conf= 100% Table→ chair~sofa sup= 50% conf= 100%
Chair → Sofa sup= 50% conf= 66.6% Sofa → Chair sup= 50% conf= 66.6% Chair~sofa → table sup= 50% conf= 100% Table→ chair ~ sofa sup= 50% conf= 100%

Table6. Test with WEFB data mining over the set FURNITURE STORE

calculated considering the fuzzy similarity between items. Association rules obtained by WEFB data mining contain fuzzy items like chair~sofa (chair and sofa can be considered similar) and which represents interesting semantic similarities not revealed by Apriori. Analyzing the additional rules obtained by WEFB data mining, we can show that WEFB data mining generates more association rules than Apriori does, with the same support and confidence parameters. As expected, the computation performance of Apriori is better than the computational performance of WEFB data mining, because WEFB data mining has a more complex structure to find fuzzy similarity items

Test with WEFB data mining over the set FURNITURE STORE, with minsup = 40 and minconf = 40,Itemsets pair above minimum support and minimum confidence rule:
Rules generated table ~desk → sofa Fuzzy sup= 86.5000 table ~desk → seat Fuzzy sup= 43.2500 wardrobe ~cabinet → se Fuzzy sup= 44.7500

4.1 Time Complexity

Say a database D, having N entries for the various transactions in it, then time taken by our algorithm is dependent on the three factors i.e. (i) The value of minimum support, (ii) Minimum Confidence and (iii) Minimum Similarity.
Hence
T = some function, f (min support, min confidence, min similarity)
As well as no. of entries N in database
Hence T = f (min support, min confidence, min similarity, N)
Since the minimum support, Confidence and Similarity values subject to change even for same database (i.e. depend on human perception (expert)) hence, we cannot define this function f. Thus to have an upper bound for the time taken by the algorithm we will redefine it as: In a transaction database, if there are |D| transactions, and these |D| transactions having each m attributes to be read, as in the algorithm. Therefore, this we can run this in O (|D|*m) time for reading the database.

For examples if we take D=5 transactions and each transactions have m=3 attributes, then the time taken by algorithm is of the order of 15. But however the major step in the discovery of frequent item sets based on user defined weighted minimum support is the process of finding power set P. As given if there are m frequent 1-item attribute sets, then the number of possible subsets obtained from this is 2^m. Hence this step can be completed is in O (2^m).
We can easily verify it as say for example if we take 100 attributes for 5 transactions then the order of it is 2^{100} that's much larger than the 500. Thus we can definitely be completed is in O(2^m).

4.2 Space Complexity

To compute the space complexity we consider the similarity value table defined by an expert as in a database D, if we have m items and each item has three attributes, as in our example database is the item id, transaction id and similarity value. Since as we increase the m items it is observed that the similarity values required for this is the order of 3xm. It means total space required for the similarity Database is in order O (3m).
This can further verified with an example as, say in our database if we consider three items Bread, Butter and Milk then we need to find the similarity values between
Bread-Butter
Bread-Milk, and
Butter-Milk i.e. three.
And as per said its order 3*3=9. So three $\leq$ 9.
Similarly if we take seven items Bread, Butter, Egg, Milk, Soap, Toothpaste and Brush then we need to find the similarity between
Bread-Butter, Bread-Milk, Bread-Egg, Bread-Soap, Bread-Toothpaste, Bread-Brush, Butter-Egg, Butter-Milk, Butter-Soap, Butter-Toothpaste, Butter-Brush, Egg-Milk, Egg-Soap, Egg-Toothpaste, Egg-Brush, Milk-Soap, Milk-Toothpaste, Milk-Brush, Soap-Toothpaste, Soap-Brush and Toothpaste-Brush i.e. Twenty-one.
And as per said order it is 3x7 = 21. So Twenty-one $\leq$ 21

Thus in terms of asymptotic notations it is right to say that the total space required for the Database is in order O (3m).

5. PERFORMANCE GRAPH

For the experimental analysis, there is the requirement of extremely large datasets, hence we used the transactions datasets of 1K 2K 3K 4K 5K with 20 items for a small grocery store. Thus we evaluated the comparison table between two methods namely with expert as well as without expert based Data mining. From the Fig 4 it is quite clear that without-expert based Data mining outperformed above the with-expert based mining.

Data Sets Used in K	With an expert Rules Generated	Without Expert Fuzzy Based Rules Generated
1 K	43	55
2 K	76	83
3 K	92	97
4 K	113	123
5 K	141	153

Table 7: Numbers of rules generated with an expert and without expert fuzzy based

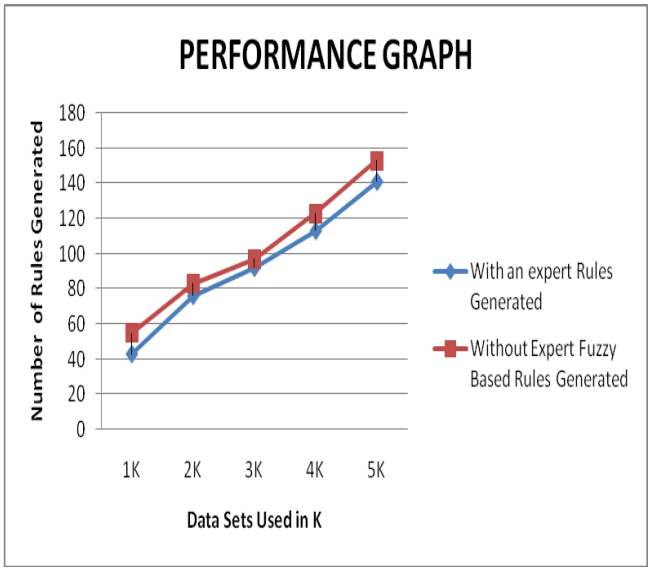

Figure4. Comparison between number of rules generated with an expert and Without expert fuzzy based

6. CONCLUSION

We have successfully designed the data mining algorithm based on its fuzzy similarity values and support and confidence values, which have been used to implement the data mining for very large data. With the creation and application of proposed work, it has become possible to discover the new hidden association rules that reflect the fuzzy similarity among data which work previously revealed by known data mining algorithm. The use of fuzzy logic concepts in fuzzy based data mining contributed to make information representation and manipulation closer to the human language, making them more understandable infact even by laymen. In data mining the general rule is the better the comprehension of the obtained knowledge, the bigger the knowledge utility. We have also discussed the data mining challenges, in which the researches are required for developing efficient and uniform data mining algorithms, software tools and techniques for very large, high dimensional and complex data.

7. FUTURE WORK

As future work, we can enhance the computational performance of fuzzy based data mining algorithm. We can also plan to define a more refined way of expressing the concepts involved in the fuzzy based data mining using fuzzy weighted association rules.

8. REFERENCES

[1] W.H. Au, K.C.C. Chan, An effective algorithm for discovering fuzzy rules in relational databases, in: Proc. IEEE Internat. Conf. Fuzzy Systems,vol. II, 1998, pp. 1314–1319.

[2] W.H. Au, K.C.C. Chan, FARM: a data mining system for discovering fuzzy association rules, in: Proc. FUZZ-IEEE'99, vol. 3, 1999, pp. 22–25.

[3] Han, J. and Kamber, M. (2001) "Data Mining - Concepts and Techniques", 1st Edition.Nova York: Morgan Kaufmann.

[4] Chen, G. and Wei, Q. (2002) "Fuzzy association rules and the extended mining algorithms",

[5] Fuzzy Sets and Systems, v. 147, n. 1-4, p. 201-228 X.Wu, C.Zhang, and S.Zhang, Mining both Positive and Negative Association Rules, Proc. Of 19th Int. Conf. on Data Machine Learning, pp.658-665,2002.

[6] T. P. Hong, K. Y. Lin and S. L. Wang, "Mining fuzzy association rules from quantitative transactions", Soft Computing, Vol. 10, No.10, pp. 925-932, 2006.

[7] M. Kaya, R. Alhajj, F. Polat and A. Arslan, "Efficient Automated Mining of Fuzzy Association Rules," *Proc. Of a*DEXA, *2002*.

[8] Gagan Kumar, Neeraj Mangla and Aakanksha Mahajan, "Improved Data Mining Based on Semantic Similarity to mine new Association Rules" CIIT International Journal in press.