AN ALGORITHM FOR GENERATING SINGLE DIMENSIONAL FUZZY ASSOCIATION RULE MINING

Rolly Intan
Faculty of Technology Industry, Informatics Engineering Department, Petra Christian University
e-mail: rintan@petra.ac.id

ABSTRACT*:* Association rule mining searches for interesting relationship among items in a large data set. Market basket analysis, a typical example of association rule mining, analyzes buying habit of customers by finding association between the different items that customers put in their shopping cart (basket). Apriori algorithm is an influential algorithm for mining frequent itemset for generating association rules. For some reasons, Apriori algorithm is not based on human intuitive. To provide a more human-based concept, this paper proposes an alternative algorithm for generating the association rule by utilizing fuzzy sets in the market basket analysis.

Keywords: fuzzy association rule mining, data mining, apriori algorithms, fuzzy sets.

INTRODUCTION

Association rule mining finds interesting association or correlation relationship among a large data set of items[1]. The discovery of interesting association rules can help in decision making process. Market basket analysis is considered as a typical example of association rule mining. In market basket analysis, customers buying habit is analyzed for finding association between different items customers put together in their shopping cart. Two different items, *a* and *b*, in an itemset are assumed to have a relation if they are purchased together in the same transaction. In this case, it can be considered that the customer buy *a* is because he/she buy *b*, and vice versa. More two items are purchased together in the same transaction, more they have stronger relation. The discovery of such associations can help retailers develop marketing strategies by gaining insight into which items are frequently purchased together by customers.

Apriori is an influential algorithm for mining frequent itemsets for Boolean association rules. Generally, based on the algorithms, 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 the number of items in a record of transaction. However, based on *human intuitive*, it should be considered that the larger number of items purchased in a transaction means that the degree of association among the items in the transaction may be lower. This concept was proposed and discussed well in[3] as concluded as follows.

"*Every element or object will have a relation (similarity) to the others if they are involved in the same group (class). They will have stronger relationship (similarity) if they are involved in more the same groups (classes). On the other hand, an increasing number of elements in a class will reduce the degree of relationship (similarity) among the element involved in the class.*"

Based on the concept, this paper proposes an alternative algorithm for improving Apriori algorithm in generating the association rule by utilizing fuzzy sets in the market basket analysis. Generating rules mining representing association between two itemsets as given in Apriori algorithms is generalized to generating rules mining representing association between two *fuzzy* itemsets. A fuzzy itemset may be represented by a meaningful fuzzy label, i.e. "soft drink", "snack", etc, where a fuzzy itemset "soft drink" may be arbitrarily given by:

$$\mu_{SoftDrink} = \{\frac{1}{Coke}, \frac{1}{Sprite}, \frac{0.6}{Beer}\}.$$

Two important formulas are introduced to calculate *support* and *confidence factor* for every association rule.

The structure of the paper is the following. In Section 2, basic concepts of Market Basket Analysis and Apriori algorithms are recalled and discussed. Section 3 as a main contribution of this paper is devoted to propose an algorithm for generating fuzzy association rules mining. Section 4 demonstrated the algorithms in an illustrative example. Finally a conclusion is given in Section 5.

MARKET BASKET ANALYSIS AND APRIORI ALGORITHM

Market basket data analysis[1], a typical example of association rule mining, analyzes buying habit of customers by finding association between the different items that customers put in their shopping cart (basket) as recorded in a transactional database. 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 as shown in Table 1.

Table 1. A Transactional Database

Trans_ID	*List of Items*
T_1	i_1, i_2, i_5
T_2	i_2, i_4
T_3	i_2, i_3
T_4	i_1, i_2, i_4
T_5	i_1, i_3
T_6	i_2, i_3
T_7	i_1, i_3
T_8	i_1, i_2, i_3, i_5
T_9	i_1, i_2, i_3
T_{10}	i_1, i_2, i_3, i_6

A transactional database may have additional information regarding the sale such as customer ID, date of transaction, etc.

In general, results of market basket analysis given a transactional database for mining frequent itemsets are in the form of Boolean association rules. In universal set of items, each item has a Boolean variable representing the presence or absence of that item. Each basket (shopping cart) can then be represented by a Boolean vector of values assigned to these variables. The Boolean vectors can be analyzed for buying patterns that reflect items that are frequently associated or purchased together represented by the Boolean association rules in which two measures, support and confidence, determine measures of rule interestingness. They respectively reflect the usefulness and certainty of discover rules. For instance, from a rule:

$A \Rightarrow B$ [support=5%, confidence=60%],

a support of 5% means that 5% of all transactions under analysis show that *A* and *B* are purchased together. A confidence 60% means that 60% of the customers who purchased *A* also bought *B*. Formally, let $\Im = \{i_1, i_2, \ldots, i_m\}$ be a universal set of items. Let ***D*** be a set of transactions where each transaction T_i is a set of items such that $T_i \subseteq \Im$. Each transaction is associated with an identifier (*trans_ID*), i.e. T_1, T_2, etc, where $T_1=\{i_1,i_2,i_3\}$, $T_2=\{i_2,i_4\}$ (see Table 2.1). Let *A* be a set of items. A transaction T_i is said to contain *A* if and only if $A \subseteq T_i$. An association rule is an implication of the form: $A \Rightarrow B$, where $A \subset \Im, B \subset \Im$, and $A \cap B \neq \emptyset$. The rule $A \Rightarrow B$ holds in the transaction set ***D*** with support *s*, where *s* is the percentage of transactions in ***D*** that contains $A \cup B$. This is taken to be the probability $P(A \cup B)$. The rule $A \Rightarrow B$ has confidence *c* in the transactions set ***D*** if c is the percentage of transactions in ***D*** containing *A* that also contain *B*. This is taken to be the conditional probability, $P(A \mid B)$. That is,

support $(A \Rightarrow B) = P(A \cup B)$,

confidence $(A \Rightarrow B) = P(A \mid B)$.

Apriori[1] is an influential algorithm in market basket analysis for mining frequent itemsets for Boolean association rules. 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. To illustrate the algorithm, first, every item in Table 1 is scanned to count its number of occurrences as shown in Table 2, where each item is a member of candidate 1-itemsets, C_1. Suppose that the minimum support, denoted by β, equals to 2, the set of frequent 1-itemsets, L_1, can be determined as given in Table 3.

Table 2. C_1

C_1	
1-itemsets	**Sup.count**
$\{i_1\}$	7
$\{i_2\}$	8
$\{i_3\}$	7
$\{i_4\}$	2
$\{i_5\}$	2
$\{i_6\}$	1

Table 3. L_1 **(β=2)**

L_1	
1-itemsets	**Sup.count**
$\{i_1\}$	7
$\{i_2\}$	8
$\{i_3\}$	7
$\{i_4\}$	2
$\{i_5\}$	2

Similarly, *L2* can be generated from *L1*, and *L3* can be generated from *L2* as given in Table 4 and Table 5, respectively. The set of frequent itemsets can be used to generate strong association rules, where strong association rules satisfy both minimum support and minimum confidence. Calculation of confidence factor can be done by the following equation:

$$confidence(A \Rightarrow B) = P(B \mid A) = \frac{Sup.count(A \cup B)}{Sup.count(A)}$$

Table 4. L_2 (β=2)

L_2	
2-itemsets	**Sup.count**
$\{i_1, i_2\}$	5
$\{i_1, i_3\}$	5
$\{i_1, i_5\}$	2
$\{i_2, i_3\}$	5
$\{i_2, i_4\}$	2
$\{i_2, i_5\}$	2

Table 5. L_3 (β=2)

L_3	
3-itemsets	**Sup.count**
$\{i_1, i_2, i_3\}$	3
$\{i_1, i_2, i_5\}$	2

The resulting association rules are shown as follows.

$i_1 \wedge i_2 \Rightarrow i_5,\quad confidence = 40\%$

$i_1 \wedge i_5 \Rightarrow i_2,\quad confidence = 100\%$

$i_2 \wedge i_5 \Rightarrow i_1,\quad confidence = 100\%$

$i_1 \Rightarrow i_2 \wedge i_5,\quad confidence = 29\%$

$i_2 \Rightarrow i_1 \wedge i_5,\quad confidence = 25\%$

$i_5 \Rightarrow i_1 \wedge i_2,\quad confidence = 100\%$

GENERATING FUZZY ASSOCIATION RULES

As mentioned in Section 1, Apriori algorithm ignored the number items in a shopping cart in determining relationship of the items. Section 2 shows steps of the algorithm that calculation of support of itemsets just count the number of occurrences of the itemsets in every record of transaction (shopping cart), without any consideration to the number of items in a record of transaction. However, based on *human intuitive*, it should be considered that the larger number of items purchased in a transaction means that the degree of association among the items in the transaction may be lower. For instance, Table 3.1 is given to understand this concept. Based of Apriori algorithm as discussed in Section 2, supports of itemsets, $\{i_1,i_2\}$ and $\{i_2,i_3\}$ are the same (equal to 3). However, this calculation is not fair considering that there is a record of transaction, (T_1), where i_1 and i_2 are purchased together without any other items. It could be said that i_1 and i_2 may have a strong relation one to each other. The customer may buy i_1 because of buying i_2 and vice versa. On the other hand, there is no case in which i_2 and i_3 are purchased together without any other items. That means that the reason why a customer buys i_3 is not only because of i_2, but also because of i_1 or i_4 (as shown in T_2), i_6, i_7, or i_8 (as shown in T_3), i_1 or i_9 (as shown in T_4). Compare to the degree of relationship between i_1 and i_2, the degree of relationship between i_2 and i_3 should be weaker. In other words, the degree of relationship between i_1 and i_2 should be higher than the degree of relationship between i_2 and i_3.

Table 6. Transactional Database

Trans_ID	*List of Item*
T_1	i_1, i_2
T_2	i_1, i_2, i_3, i_4
T_3	i_2, i_3, i_6, i_7, i_8
T_4	i_1, i_2, i_3, i_9

To improve Apriori algorithm, a new algorithm is proposed by considering that every item will have a relation (similarity) to the others if they are purchased together in the same record of transaction. They will have stronger relationship if they are purchased in more the same transactions. On the other hand, increasing number of items in a transaction will reduce the degree of relationship among the items involved in the transaction.

The proposed algorithm is given in the following steps:

Step-1:

Determine $\delta \in \{2,3,\ldots,\aleph_n\}$ (maximum item threshold). δ is a threshold to determine maximum number of items in a transaction by which the transaction may or may not be considered in the process of generating rules mining. In this case, the process just considers all transactions with the number of items in the transactions less than or equal to δ. Formally, let **D** be a universal set of transactions. $\mathbf{M} \subseteq \mathbf{D}$ is considered as a subset of qualified transactions for generating rules mining that the number of items in its transactions is no greater than δ as defined by:

$$\mathbf{M}=\{T \mid card(T) \le \delta,\ T \in \mathbf{D}\}, \quad (1)$$

where card(T) is the number of items in transaction T.

Step-2:

Set k=1, where k is an index variable to determine the number of combination items in itemsets called k-itemsets.

Step-3:
Determine minimum support for *k*-itemsets, denoted by $\beta_k \in (0,|M|]$ as a minimum *threshold* of a combination *k* items appearing in the whole qualified transactions, where |**M**| is the number of qualified transactions. Here, β_k may have different value for every *k*.

Step-4:
Construct every candidate *k*-itemset, I^k, as a fuzzy set on set of qualified transactions, **M**.
A *fuzzy membership function,* μ, is a mapping: $\mu_{I^k} : M \rightarrow [0,1]$ as defined by:

$$\mu_{I^k}(T) = \inf_{i \in I^k}\{\frac{\eta_T(i)}{card(T)}\}, \quad \forall T \in M \qquad (2)$$

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:

$$\eta_T(i) = \begin{cases} 1, & i \in T \\ 0, & \text{otherwise} \end{cases} \qquad (3)$$

such that if an item, *i*, is an element of T then $\eta_T(i)=1$, otherwise $\eta_T(i)=0$.

Step-5:
Calculate *support* for every (candidate) *k*-itemset using the following equations:

$$support\ (I^k) = \sum_{T \in M} \mu_{I^k}(T) \qquad (4)$$

M is the set of qualified transactions as given in (1); It can be proved that (4) satisfied the following property: $\sum_{i \in \Im} support(i) = |M|$.
For *k*=1, I^k can be considered as a single item.

Step-6:
I^k will be stored in the set of frequent *k*-itemsets, L_k if and only if *support* $(I^k) \geq \beta_k$.

Step-7:
Set *k*=*k*+1, and if $k > \delta$, then go to Step-9.

Step-8:
Looking for possible/candidate *k*-itemsets from L_{k-1} by the following rules: A k-itemset, I^k, will be considered as a candidate *k*-itemset iff I^k satisfied:

$$\forall F \subset I^k, \ |F| = k-1 \Rightarrow F \in L_{k-1}$$

For example, $I^k=\{i_1, i_2, i_3, i_4\}$ will be considered as a candidate 4-itemset, iff: $\{i_1, i_2, i_3\}$,$\{i_1, i_2, i_4\}$,$\{i_1, i_3, i_4\}$ and $\{i_2, i_3, i_4\}$ are in L_3. If there is not found any candidate k-itemset then go to Step-9. Otherwise, the process is going to Step-3.

Step-9:
Similar to Apriori Algorithm, confidence of an association rule mining, $A \Rightarrow B$, can be calculated by the following equation:

$$cf(A \Rightarrow B) = P(B \mid A) = \frac{support(A \cup B)}{support(A)} \qquad (5)$$

where $A, B \subseteq \Im$.
It can be followed that (5) can be also represented by:

$$cf(A \Rightarrow B) = \frac{\sum_{T \in M} \inf_{i \in A \cup B}(\mu_i(T))}{\sum_{T \in M} \inf_{i \in A}(\mu_i(T))} \qquad (6)$$

where *A* and *B* are any *k*-itemsets in L_k. (Note: $\mu_i(T) = \mu_{\{i\}}(T)$, for simplification). Therefore, support of an itemset as given by (4) can be also expressed by:

$$support\ (I^k) = \sum_{T \in M} \inf_{i \in I^k}(\mu_i(T)) \qquad (7)$$

In general, let $X_1, X_2, \cdots X_m \subseteq \Im$:

$$\begin{aligned} support(X_1 \cup \cdots \cup X_m) &= \sum_{T \in M} \inf_{j=1}^{m}\{\inf_{i \in X_j}(\mu_i(T))\} \\ &= \sum_{T \in M} \inf_{i \in \mathbf{X}}(\mu_i(T)), \end{aligned} \qquad (8)$$

where $\mathbf{X} = X_1 \cup X_2 \cup \cdots \cup X_m$.
Obviously, (6) can be directly generated from (5) by (8).

The itemsets in the previous discussion may be regarded as *crisp* itemsets, where the itemsets are crisp subsets of items. Every crisp itemset can be generalized to be a *fuzzy* itemset symbolized by a meaningful fuzzy label. A fuzzy itemset, *X*, is a fuzzy set on the set of items, $\Im$ characterized by a *membership function*, λ, where $\lambda_x : \Im \rightarrow [0,1]$. In general, (8) can be extended to provide a more generalized formula for utilizing fuzzy itemsets. Let $X_1, X_2, \cdots X_m$ be fuzzy sets on $\Im$. Support of union all the fuzzy sets is defined by:

$$support(X_1 \cup \cdots \cup X_m) = \sum_{T \in M} \inf_{j=1}^{m}\{\inf_{i \in \Phi_j}(\lambda_{X_j}(i) \cdot \mu_i(T))\}, \qquad (9)$$

where $\Phi_j = \{i \mid \lambda_{X_j}(i) > 0\}$.

Boolean association rules can be generalized to be fuzzy association rules by representing association between two *fuzzy* itemsets instead of two *crisp* itemsets. Let *A* and *B* are two fuzzy itemsets or two fuzzy set on set of items such as $A, B \in \mathcal{F}(\Im)$ (where $\mathcal{F}(\Im)$ is a fuzzy power set on the set of items), from

(5) and (9), the calculation of confidence rule as given in (6) must be extended to be the following equation:

$$cf(A \Rightarrow B) = \frac{\sum_{T \in M} \min(\inf_{i \in \Phi_A}(\lambda_A(i) \cdot \mu_i(T)), \inf_{i \in \Phi_B}(\lambda_B(i) \cdot \mu_i(T)))}{\sum_{T \in M} \inf_{i \in \Phi_A}(\lambda_A(i) \cdot \mu_i(T))} \quad (10)$$

where $\Phi_A = \{i | \lambda_A(i) > 0\}$ and $\Phi_B = \{i | \lambda_B(i) > 0\}$. A *fuzzy membership function* is a mapping: $\lambda_A, \lambda_B : \Im \rightarrow [0,1]$. On the other hand, it can be easily verified that (10) will change to (6) if *A* and *B* are two crisp itemsets in which $\lambda_A, \lambda_B : \Im \rightarrow \{0,1\}$. In other words, (10) is a generalization of (6).

AN ILLUSTRATIVE EXAMPLE

An illustrative example is given to understand well the concept of the proposed algorithm and how the process of the generating fuzzy association rule mining is performed step by step. The process is started from a given transactional database as shown in Table 1.

Step-1:
Suppose that δ arbitrarily equals to 3; that means qualified transaction is regarded as a transaction with no more than 3 items purchased in the transaction. Result of this step is a set of qualified transaction as seen in Table 4.1, where **M**=$\{T_1, T_2, T_3, T_4, T_5, T_6, T_7, T_9\}$.

Table 7. A Qualified Data Transaction (M)

Trans_ID	*List of Items*
T_1	I_1, i_2, i_5
T_2	I_2, i_4
T_3	I_2, i_3
T_4	I_1, i_2, i_4
T_5	I_1, i_3
T_6	I_2, i_3
T_7	I_1, i_3
T_9	I_1, i_2, i_3

Step-2:
The process is started by looking for support of 1-itemsets for which *k* is set equal to 1.

Step-3:
Since δ=3, then $k \in \{1,2,3\}$. It is arbitrarily given $\beta_1 = \beta_2 = 0.5$, $\beta_3 = 0.2$. That means the system just considers support of *k*-itemsets that is greater than 0.5, for k=1,2, and greater than 0.2, for k=3.

Step-4:
Every *k*-itemset is represented as a fuzzy set on set of qualified transactions as given by the following results:

<u>1-itemsets:</u>
$\{i_1\}=\{0.33/T_1, 0.33/T_4, 0.5/T_5, 0.5/T_7, 0.33/T_9\}$,
$\{i_2\}=\{0.33/T_1, 0.5/T_2, 0.5/T_3, 0.33/T_4, 0.5/T_6, 0.33/T_9\}$,
$\{i_3\}=\{0.5/T_3, 0.5/T_5, 0.5/T_6, 0.5/T_7, 0.33/T_9\}$,
$\{i_4\}=\{0.5/T_2, 0.33/T_4\}$,
$\{i_5\}=\{0.33/T_1\}$.
From Step-5 and Step-6, $\{i_5\}$ cannot be considered for further process because *support*($\{i_5\}$)< β_1.

<u>2-itemsets:</u>
$\{i_1, i_2\}=\{0.33/T_1, 0.33/T_4, 0.33/T_9\}$,
$\{i_2, i_4\}=\{0.5/T_2, 0.33/T_4\}$,
$\{i_2, i_3\}=\{0.5/T_3, 0.5/T_6, 0.33/T_9\}$,
$\{i_1, i_4\}=\{0.33/T_4\}$,
$\{i_1, i_3\}=\{0.5/T_5, 0.5/T_7, 0.33/T_9\}$.
From Step-5 and Step-6, $\{i_1, i_4\}$ cannot be considered for further process because *support*($\{i_1, i_4\}$)< β_2.

<u>3-itemsets:</u>
$\{i_1, i_2, i_3\}=\{0.33/T_9\}$,

Step-5:
Support of each k-itemset is calculate as given in the following results:
<u>1-itemsets:</u> <u>2-itemsets</u>
support($\{i_1\}$) = 1.99, *support*($\{i_1, i_2\}$)=0.99
support($\{i_2\}$) = 2.49, *support*($\{i_2, i_4\}$)=0.83
support($\{i_3\}$) = 2.33, *support*($\{i_2, i_3\}$)=1.33
support($\{i_4\}$) = 0.83, *support*($\{i_1, i_4\}$)=0.33
support($\{i_5\}$) = 0.33, *support*($\{i_1, i_3\}$)=1.33

<u>3-itemsets:</u>
support($\{i_1, i_2, i_3\}$)=0.33

Table 8. $L_1(\beta_1=0.5)$

L_1	
1-itemsets	**Support**
$\{i_1\}$	1.99
$\{i_2\}$	2.49
$\{i_3\}$	2.33
$\{i_4\}$	0.83

Table 9. $L_2(\beta_2=0.5)$

L_2	
2-itemsets	**Support**
$\{i_1, i_2\}$	0.99
$\{i_2, i_4\}$	0.83
$\{i_2, i_3\}$	1.33
$\{i_1, i_3\}$	1.33

Table 10: $L_2(\beta_3=0.2)$

L_2	
2-itemsets	**Support**
$\{i_1, i_2, i_3\}$	0.33

Step-6:
From the results as performed by Step-4 and 5, the sets of frequent 1-itemsets, 2-itemsets and 3-itemsets are given in Table 8, 9 and 10, respectively.

Step-7:
This step is just for increment the value of k in which if $k > \delta$, then the process is going to Step-9.

Step-8:
This step is looking for possible/candidate *k*-itemsets from L_{k-1}. If there is no any more candidate k-itemset then go to Step-9. Otherwise, the process is going to Step-3.

Step-9:
The step is to calculate every confidence of each possible association rules as follows:

$$cf(i_1 \Rightarrow i_2) = \frac{Support(\{i_1, i_2\})}{Support(\{i_1\})} = \frac{0.99}{1.99} = 0.5,$$

$$cf(i_2 \Rightarrow i_4) = \frac{Support(\{i_2, i_4\})}{Support(\{i_2\})} = \frac{0.83}{2.49} = 0.33,$$

$$\vdots$$

$$cf(i_1 \wedge i_2 \Rightarrow i_3) = \frac{Support(\{i_1, i_2, i_3\})}{Support(\{i_1, i_2\})} = \frac{0.33}{0.99} = 0.33,$$

$$cf(i_1 \Rightarrow i_2 \wedge i_3) = \frac{Support(\{i_1, i_2, i_3\})}{Support(\{i_1\})} = \frac{0.33}{1.99} = 0.16,$$

$$\vdots$$

Let a fuzzy association rule represents association between two *fuzzy* itemsets, *A* and *B*, where *A* and *B* are two fuzzy sets on set of items as given by $\mu_A=\{0.5/i_1, 1/i_2\}$ and $\mu_B=\{1/i_2, 0.5/i_3\}$, respectively. Confidence of the fuzzy association rule is calculated by (10) as follows. First, from *A* and *B*, Φ_A and Φ_B can be determined by $\Phi_A = \{i_1, i_2\}$ and $\Phi_B = \{i_2, i_3\}$, respectively.

$$cf(A \Rightarrow B) = \frac{0.16}{0.16 + 0.16 + 0.16} = 0.33$$

Implementation of the proposed algorithm had been partially experimented by developing a software in[6] using Borland Delphi 7 and MS Access 2003, where tested transactional database was taken from a big supermarket for a month. Outputs of the software such as association rules, association degree of rules, and support of itemsets is useful to help manager to decide market policies.

CONCLUSION

This paper introduced an algorithm for generating fuzzy association rules mining as a generalization of Boolean association rule. The algorithm is based on the concept that the larger number of items purchased in a transaction means the lower degree of association among the items in the transaction. Based on the concept, two new formulas of calculating degree of support and confidence were proposed utilizing the fuzzy set theory. Moreover, to generalize Boolean association rules, the concept of fuzzy itemsets was discussed in order to introduce the concept of fuzzy association rules. Two generalized formulas were also proposed in the relation to the fuzzy association rules. Finally, an illustrated example was given to clearly demonstrate and understand steps of the algorithm.

REFERENCES

1. J. Han, M. Kamber, *Data Mining: Concepts and Techniques*, The Morgan Kaufmann Series, 2001.
2. G. J. Klir, B. Yuan, *Fuzzy Sets and Fuzzy Logic: Theory and Applications*, New Jersey: Prentice Hall, 1995.
3. R. Intan, M. Mukaidono, "Toward a Fuzzy Thesaurus Based on Similarity in Fuzzy Covering", *Australian Journal of Intelligent Information Processing,* Vol.8 No.3, pp. ~132-139, 2004.
4. R. Intan, M. Mukaidono., "Generating Fuzzy Thesaurus by Degree of Similarity in Fuzzy Covering", *Proceedings of ISMIS 2003*, LNAI 2871, Springer-Verlag, pp. 427-432, 2003.
5. R. Intan, M. Mukaidono, "A Proposal of Fuzzy Thesaurus Generating by Fuzzy Covering", *Proceedings of NAFIPS 2003*, IEEE Press, pp. 167-172, 2003.
6. O. P. Gunawan, *Perancangan dan Pembuatan Aplikasi Data Mining dengan Konsep Fuzzy c-Covering untuk Membantu Analisis Market Basket pada Swalayan X*, (in Indonesian) Final Project, 2004.
7. L. A. Zadeh, "Fuzzy Sets and systems," *International Journal of General Systems*, Vol. 17, pp. 129-138, 1990.
8. Gregorius S. Budhi, Resmana Lim, O. P. Gunawan, "Penggunaan Metode Fuzzy c-Covering untuk Analisa Market Basket pada Supermarket", *Jurnal Informatika* , Vol. 6, No. 1, Mei 2005.