International Journal of Scientific Engineering and Technology (ISSN : 2277-1581)
www.ijset.com, Volume No.1, Issue No.4, pg : 35-38 01 Oct. 2012

Association Rules Mining using Modified Genetic Algorithm

Ms. Ruchi Saxena, Prof. Shailendra Shrivastava, Prof. Abhishek Mathur

Department of Information Technology, S.A.T.I. Vidisha (M.P.) India
Email: ruchi.it@gmail.com

Abstract: ***the association rules are generally used to find out the relations between different data entities in given data set. Practically the data set can contain a large number of data entities which forms a large number of rules set but for the understanding of efficient relation it is needed that rules should be kept as minimum as possible without losing the useful information this is a difficult task and can be seen as optimization problem. In this paper we are presenting an improved genetic algorithm based association rule mining & the results shows that the proposed algorithm outperforms the genetic algorithm in terms of time and quality.***

Keywords: Association rule mining, genetic algorithm, Apriori algorithm.

1. Introduction

The association rule mining is applied in many applications of data mining where the relations between variables need to be analyzed & where rules has to be retrieved from the available transactions. In data mining, association rule learning/mining is a popular and well researched method for discovering interesting relations between variables in large databases. Piatetsky-Shapiro [1] describes analyzing and presenting strong rules discovered in databases using different measures of interestingness. Based on the concept of strong rules, Rakesh Agrawal et al. [2] introduced association rules for discovering regularities between products in large scale transaction data recorded by point-of-sale (POS) systems in supermarkets. For example, the rule found in the sales data of a supermarket would indicate that if a customer buys onions and potatoes together, he or she is likely to also buy hamburger meat. Such information can be used as the basis for decisions about marketing activities such as, e.g., promotional pricing or product placements. In addition to the above example from market basket analysis association rules are employed today in many application areas including Web usage mining, intrusion detection and bioinformatics. As opposed to sequence mining, association rule learning typically does not consider the order of items either within a transaction or across transactions.

2. Association Rules

Association rules are if/then statements that help uncover relationships between seemingly unrelated data in a relational database or other information repository. An example of an association rule would be "If a customer buys a dozen eggs, he is 80% likely to also purchase milk."

2. Association Rules

An association rule has two parts, an antecedent (if) and a consequent (then). An antecedent is an item found in the data. A consequent is an item that is found in combination with the antecedent. Association rules are created by analyzing data for frequent if/then patterns and using the criteria support and confidence to identify the most important relationships. Support is an indication of how frequently the items appear in the database. Confidence indicates the number of times the if/then statements have been found to be true. In data mining, association rules are useful for analyzing and predicting customer behavior. They play an important part in shopping basket data analysis, product clustering, catalog design and store layout.

The idea of association rules is originated since the market–basket where you want to find dependence between two items X and Y . A good example is the way “A client who wishes to buy products X_1 and X_2 will also buy product Y”. An association rule is an implication $X \rightarrow Y$, where X is the antecedent (a conjunction of conditions) and Y is the consequent (predict class). Besides, X and Y are disjoint sets of items, i.e., $X \cap Y = \emptyset$ [5].

There are three general characteristics that discovery of rules must satisfy; to have specifically a high precision prediction,

to be understandable and to be interesting [3]. A measure to predict the association rule precision X →Y is the confidence. This measures the reliability of interference made by the rule which is defined like:

$$C = \frac{|X \cup Y|}{X} \ldots\ldots\ldots\ldots\ldots\ldots (1)$$

Where |X| is the number of examples that satisfies every condition in the antecedent X and |X∪Y| is the number of examples both of which satisfy the antecedent X and it has the class predicted by the consequent Y. But the confidence favors the rules over fitting the data [3]. Due to this it is necessary to determine the way a rule is applicable in dataset, such as, support. It is defined as:

$$C = \frac{|X \cup Y|}{N} \ldots\ldots\ldots\ldots\ldots\ldots (2)$$

Where, N is the total number of examples. Support is often used to eliminate non interesting rules [5] .A measure to determine a rule interestingness is to find surprisingness of an attribute based on each attribute information gain [4].

2.1 Apriori Algorithm

It is a classic algorithm for learning association rules. Apriori is designed to operate on databases containing transactions (for example, collections of items bought by customers, or details of a website frequentation). Other algorithms are designed for finding association rules in data having no transactions (Winepi and Minepi), or having no timestamps (DNA sequencing) [7].

As is common in association rule mining, given a set of itemsets (for instance, sets of retail transactions, each listing individual items purchased), the algorithm attempts to find subsets which are common to at least a minimum number C of the itemsets. Apriori uses a "bottom up" approach, where frequent subsets are extended one item at a time (a step known as candidate generation), and groups of candidates are tested against the data. The algorithm terminates when no further successful extensions are found.

The purpose of the Apriori Algorithm is to find associations between different sets of data. It is sometimes referred to as "Market Basket Analysis". Each set of data has a number of items and is called a transaction. The output of Apriori is sets of rules that tell us how often items are contained in sets of data.

Algorithm Pseudocode

The pseudocode for the algorithm is given below for a transaction database T, and a support threshold of ε. Usual set theoretic notation is employed, though note that T is a multiset. C_k is the candidate set for level k. Generate() algorithm is assumed to generate the candidate sets from the large itemsets of the preceding level, heeding the downward closure lemma. Count[c] accesses a field of the data structure that represents candidate set c, which is initially assumed to be zero. Many details are omitted below, usually the most important part of the implementation is the data structure used for storing the candidate sets, and counting their frequencies.

```
Apriori(T, ε)
    L_1 ← { large 1-itemsets }
    k ← 2
    while L_{k-1} ≠ ∅
        C_k ← {c | c ∈ a ∪ {b} ∧ a ∈ L_{k-1} ∧ b ∈ ⋃ L_{k-1} ∧ b ∉ a}
        for transactions t ∈ T
            C_t ← {c | c ∈ C_k ∧ c ⊆ t}
            for candidates c ∈ C_t
                count[c] ← count[c] + 1
        L_k ← {c | c ∈ C_k ∧ count[c] ≥ ε}
        k ← k + 1
    return ⋃_k L_k
```

3. Measures of Association rules

To select interesting rules from the set of all possible rules, constraints on various measures of significance and interest can be used. The best-known constraints are minimum thresholds on support and confidence.

3.1 Support

Support, sometimes referred to as the cover, is the number of data points (customers, transactions, etc.) that meet a set of rules and/or assumptions. If we do a market basket analysis and find that customers who buy milk also buy cereal, the support would be the number of customers in the sample set where this holds true [6].

"The *support* supp(X) of an itemset X is defined as the proportion of transactions in the data set which contain the itemset".

3.2 Confidence

Since a rule with a support of 900 looks good when the sample size is 1,000 and not so good when the sample size is 1,000,000, we need a way to easily figure out whether or not our support is significant. Confidence is a ratio that takes the support number and divides it by the number of instances where the rule may hold true (or to be more exact - where the antecedent of our rule holds true). For instance, in our milk/cereal example above, confidence would be the total number of customers who bought milk and cereal divided by the total number of customers that bought milk [6].

"The *confidence* of a rule is defined Conf(X => Y) = supp(X => Y)/supp(X)".

4. Genetic Algorithm

Genetic algorithms are methods based on biological mechanisms, such as, Mendel's laws and Darwin's fundamental principle of natural selection. The most important biological terminology used in a genetic algorithm is [5]:

- The chromosomes are elements on which the solutions are built (individuals).
- Population is made of chromosomes.
- Reproduction is the chromosome combination stage. Mutation and crossover are reproduction methods.
- Quality factor (fitness) is also known as performance index, it is an abstract measure to classify chromosomes.
- The evaluation function is the theoretical formula to calculate a chromosome's quality factor.

Genetic algorithms simulate a population evolution process. A problem represented by individuals since a population of solutions, operators which simulate interventions about the genome such as crossover or mutation in order to achieve a population of solutions increasingly adapted to the problem. This adaptation is evaluated by the quality factor (fitness).

5. Proposed Algorithm

5.1 Modified Genetic Algorithm

The reproduction in simple genetic algorithm is considered for determining which chromosomes will be chosen as the basis of the next generation. Generating populations from only two parents may cause loss of the best chromosome from the last population. It uses the chromosome of the previous generation if the present generation's best chromosome having less fitness values than previous it also mutate just after successive repetition of similar fitness values.

5.2 Complete Algorithm

The algorithm can be performed in following steps:

1. Load the given dataset and convert the data into numeric values.

2. Apply Apriori algorithm to generate the association rules.

3. Define the fitness function for specific support & confidence values.

4. Generate N chromosomes randomly depending upon dataset length.

5. Iterate it & select the rule which satisfies the fitness conditions (according to genetic algorithm).

6. Simulation Results

The proposed algorithm is simulated for different length of dataset and variables and the results are shown below

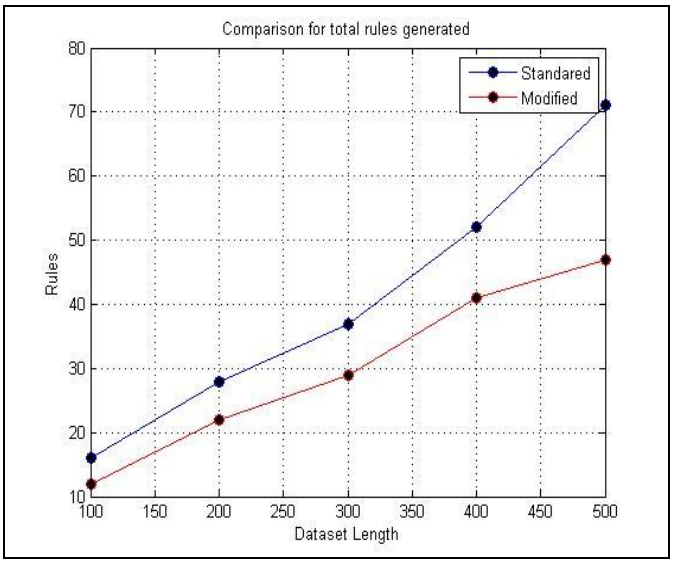

Figure 1: Comparison for total rules generated

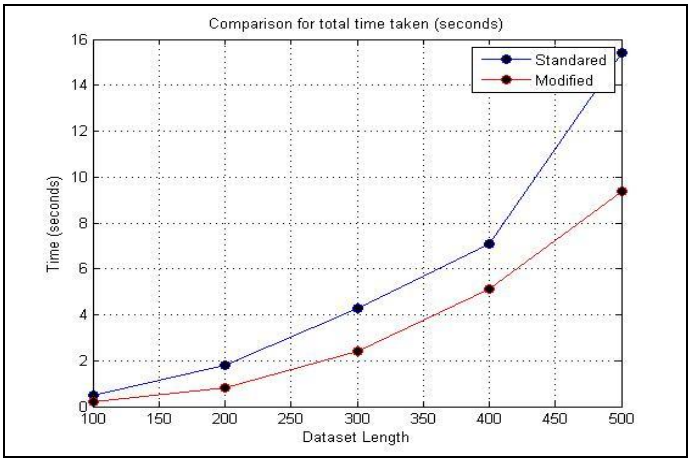

Figure 2: Comparison for total time taken

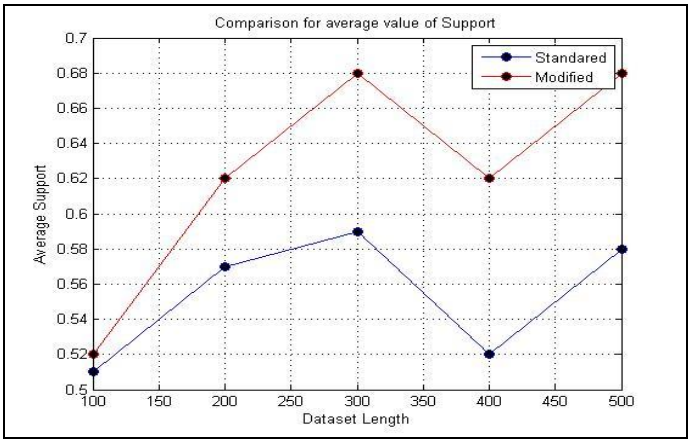

Figure 3: Comparison for average value of Support.

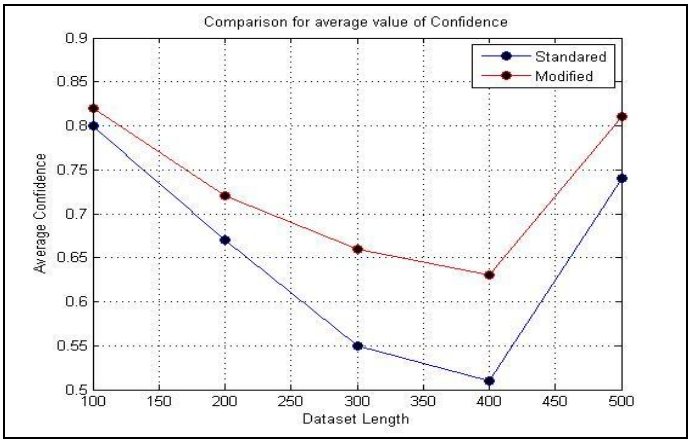

Figure 4: Comparison for average value of Confidence.

7. Conclusion

In this paper a modified genetic algorithm based association rule mining is presented the modification is performed for selecting the best chromosomes even from previous generations which should give much better performance & the non periodic mutation could reduce the unnecessary iteration resulting time reduction. The simulation results show that the expected goals are achieved because the rules selected by proposed method having much better (10%) support & confidence values with 60% reduction in processing time.

References

[1] Piatetsky-Shapiro, Gregory; and Frawley, William J., "Discovery, analysis, and presentation of strong rules" Knowledge Discovery in Databases, AAAI/MIT Press, Cambridge, MA (1991).

[2] Agrawal, Rakesh; Imielinski, Tomasz; Swami, Arun, "Mining Association Rules Between Sets of Items in Large Databases", SIGMOD Conference 1993:207-216.

[3] Freitas, Alex A. A Survey of Evolutionary Algorithms for Data Mining and Knowledge Discovery. In Advances in evolutionary computing. Eds. Ashish Ghosh and Shigeyoshi Tsutsui,Springer–Verlag New York, NY, USA, 819–845. 2003.

[4] Freitas, Alex A. On rule interesting measures. Knowledge Based System, 12(5): 309–315, 1999.

[5] Wilson Soto and Amparo Olaya–Benavides," A Genetic Algorithm for Discovery of Association Rules".

[6]http://istobe.com/blog/2008/07/22/defining-success-lift-support-and-confidence/

[7] Rakesh Agrawal and Ramakrishnan Srikant, "Fast algorithms for mining association rules in large databases",Proceedings of the 20th International Conference on Very Large Data Bases, VLDB, pages 487-499, Santiago, Chile, September 1994.