ISSN-2320-7884 (ONLINE)
ISSN-2321-0257 (PRINT)

Association Rule Mining Based on Apriori Algorithm

POTU NARAYANA
ASSISTANT PROFESSOR, MADHIRA INSTITUTE OF TECHNOLOGY & SCIENCES, KODADA, NALGONDA, ANDHRA PRADESH

Abstract:

In Data Mining, the usefulness of association rules is strongly limited by the huge amount of delivered rules. This work is pruning of association rules, generated by mining process. Association Rule Mining means discovering interesting patterns with in large databases. Association rules are used in many application areas such as market base analysis, web log analysis, protein substructures. When the number of association rules become large, it becomes less interesting to the user. It is crucial to help the decision-maker with an efficient post processing step in order to select interesting association rules throughout huge volumes of discovered rules. This motivates the need for association analysis. This paper proposes a new interactive approach to prune and filter discovered rules. First, we propose to integrate user knowledge in the processing task. Second, we propose to use the Apriori Algorithm along with Rule Schema formalism extending the specifications to obtain association rules from knowledge base. We use the high confidence value based classifier for classifying the given text document to that particular domain. Proposed system scales linearly with the number of transactions. In addition, the execution time decreases a little as the number of items in the database increases. As the average transaction size increases (while keeping the database size constant), the execution time increases only gradually. We believe that the results of this approach will help decision maker for making important decisions.

***Keywords* - Association Rules, Association Rule Mining, Pruning, Filters.**

I INTRODUCTION

One important area in data mining is concerned with the discovery of interesting association rules. Association Rules describes relationship within set of coexisting data. Association rule mining is considered as one of the most important tasks in Knowledge Discovery in Databases. It aims at discovering implicative tendencies that can be valuable information for the decision-maker. An association rule a → b implies the presence of the item set b when an item set an occurs in a database transaction. Furthermore, valuable information is often represented by those rare-low support and discovered association rules are unexpected which are surprising to the user.

Apriori algorithm extracts all association rules satisfying minimum thresholds of support and Confidence. The association rules are divided into five categories: trivial, known and correct, unknown and correct, known and incorrect, unknown and incorrect. When applying association mining to real datasets, a major obstacle is that often a huge number of rules are generated even with very reasonable support and confidence. Post-processing can be efficiently integrated with existing rule reduction techniques to construct a concise, high-quality, and user-specific association rule set.

As a result, it is necessary to bring the support threshold low enough in order to extract valuable information. Unfortunately, the lower the support is, the larger the volume of rules becomes, making it intractable for a decision-maker to analyze the mining result.

This paper proposes a new interactive approach to prune and filter discovered rules. First, we propose to integrate user knowledge in the processing task. Second, we propose to use the Apriori algorithm along with Rule Schema formalism extending the specifications to obtain association rules from knowledge base. We use the high confidence value based classifier for classifying the given text document to that particular domain. In proposed system, Rule Schemas bring the expressiveness of association rules by combining item constraints. The rule schema filter is based on operators applied over rule schemas allowing the user to perform several actions over the discovered rules. We propose two important operators: pruning and filtering operators. The filtering operator is composed of three different operators: conforming, unexpectedness, and exception. Proposed system scales linearly with the number of transactions. In addition, the execution time decreases a little as the number of items in the database increases. As the average transaction size increases (while keeping the database size constant), the execution time increases only gradually. We believe that the results of this approach will help decision maker for making important decisions.

II RELATED WORK

According to Ping Chen et al [1], post processing can be efficiently integrated with existing rule reduction techniques to construct a concise, high-quality, and user-specific association rule set. Also Claudia et al. [2] have proposed the usefulness of association rules by overcoming an interactive approach to prune and filter discovered rules. This approach has produced sets of rules, and number of ways to reduce the rules and has integrated domain expert knowledge in the post processing step. The quality of the filtered rules was also validated by the domain expert at various points in the interactive process.

Alaa Al Deen et al[3], has used the association rule mining in classification approach and experimental study against 13 UCI data sets is presented to evaluate and compare traditional and association rule based classification techniques with regards to classification accuracy, number of derived rules, rules features and processing time.

Faten et al. [4], has proposed an algorithm for building ontology via set of rules generated by rule based learning system. This algorithm has extracted the rules generated from the original dataset in developing ontology elements. Domain ontology enhances the mining results of Association Rules, which also reduce the number of generated association rules. The adopted model is based on generalization and specialization processes in which the rules are filtered by metrics based on the coverage and confidence indicators.

Hongyu Zhang et al. [5] have proposed the graphical representations of ontology‘s to help define some complexity measures intuitively. They have classified these metrics into two sets: one for measuring the overall design complexity of an ontology (ontology- level metrics), and the other for measuring the complexity of internal structure classlevel metrics).

Li [6] proposed optimal rules sets, defined with respect to interestingness metric. An optimal rule set contains all rules except those with no greater interestingness than one of its more general rules. A set of reduction techniques for redundant rules was proposed and implemented in [6]. The developed techniques are based on the generalization/specification of the antecedent/consequent of the rules and they are divided in methods for multi antecedent rules and multi consequent rules.

Ng et al. [7] proposed architecture for exploratory mining of rules. the lack of user exploration and control, the rigid notion of relationship, and the lack of focus. In order to overcome these problems, Ng et al. proposed a new query language called Constrained Association Query and they pointed out the importance of user feedback and user flexibility in choosing interestingness metrics.

Pasquier et al. [8] proposed the Close algorithm in order to extract association rules. Close algorithm is based on a new mining method: pruning of the closed set lattice (closed itemset lattice) in order to extract frequent closed itemsets. Association rules are generated starting from frequent itemsets generated from frequent closed itemsets.

III APRIORI BASIC STEPS

Apriori is a seminal algorithm for finding frequent itemsets using candidate generation. It is characterized as a level-wise complete search algorithm using anti-monotonicity of item sets, “if an item set is not frequent, any of its superset is never frequent”. Apriori first scans the database and searches for frequent itemsets of size 1 by accumulating the count for each item and collecting those that satisfy the minimum support requirement. It then iterates on the following three steps and extracts all the frequent itemsets.

1. Generate *Ck*+1, candidates of frequent itemsets of size *k* +1, from the frequent itemsets of size *k*.
2. Scan the database and calculate the support of each candidate of frequent itemsets.
3. Add those itemsets that satisfies the minimum support requirement to *Fk*+1.

IV BASIC MECHANISM

Algorithms for discovering large itemsets make multiple passes over the data. In the first pass, we count the support of individual items and determine which of them are large, i.e. have minimum support. In each subsequent pass, we start with a seed set of itemsets found to be large in the previous pass. We use this seed set for generating new potentially large itemsets, called candidate itemsets, and count the actual support for these candidate itemsets during the pass over the data. At the end of the pass, we determine which of the candidate itemsets are actually large, and they become the seed for the next pass. This process continues until no new large item sets are found.

V PROPOSED SCHEME

In proposed system, the first pass of the algorithm simply counts item occurrences to determine the large l-itemsets. A subsequent pass, say pass k, consists of two phases. First, the large itemsets Lk-1 found in the (k-1) pass are used to generate the candidate itemsets ck, using the apriorigen function. Next, the database is scanned and the support of candidates in ck is counted. For fast counting, we need to efficiently determine the candidates in ck that are contained in a given transaction t as shown in fig 1.

```
L1 = {large 1-itemsets};
for ( k = 2; Lk-1≠ 0; k++ ) do begin
Ck = apriori-gen(Lk-1); // New candidates
for all transactions t Є D do begin
Ct = Subset(Ck, t);
forall candidates c Є Ct do
c.count++;
end
Lk = {c Є Ck | C.count ≥minsup}
end
Answer = Uk Lk;
```

Figure 1: Algorithm Apriori

a) Apriori Candidate Generation:

Apriori-gen function takes as argument Lk- 1, the set of all large (k - 1)-itemsets. It returns a superset of the set of all large k-itemsets. The function works as follows: First, in the join step, we join Lk-1with Lk-1:

```
Insert into Ck
Select p.iteml, p.item2 . . . p.itemk-1, q.itemk-1
From Lk-1 p, Lk-1 q where p.iteml = q.iteml, . . ., pitemk-2 = q.itemk-2, p.itemk-1 < q.itemk-1;
```

Next, in the prune step, we delete all itemsets c Є ck such that some (k-1)-subset of c is not in Lk-1:

```
For all items sets c Є Ck do
For all (k-l)-subsets s of c do
If (s ∉ Lk-1) then
Delete c from ck;
```

In pass k of these algorithms, a database transaction t is read and it is determined which of the large item sets in Lk-i are present in t. Each of these large itemsets I is then extended with all those large items that are present in t and occur later in the lexicographic ordering than any of the items in 1.

b) Applying rule schema:

Rule Schemas bring the expressiveness of association rules by combining item constraints. The rule schema filter is based on operators applied over rule schemas allowing the user to perform several actions over the discovered rules. We propose two important operators: pruning and filtering operators. The filtering operator is composed of three different operators: conforming, unexpectedness, and exception.

Pruning: The pruning operator allows to the user to remove families of rules that he/she considers uninteresting. In databases, there exist, in most cases, relations between items that we consider obvious or that we already know. Thus, it is not useful to find these relations among the discovered associations. The pruning operator applied over a rule schema, P(RS), eliminates all association rules matching the rule schema. To extract all the rules matching a rule schema, the conforming operator is used.

Conforming: The conforming operator applied over a rule schema, C (RS), confirms an implication or finds the implication between several concepts. As a result, rules matching all the elements of a non-implicative rule schema are filtered. For an implicative rule schema, the condition and the conclusion of the association rule should match those of the schema.

Unexpectedness with a higher interest for the user, the unexpectedness operator U(RS) proposes to filter a set of rules with a surprise effect for the user. This type of rules interests the user more than the conforming one since, generally, a decision maker searches to discover new knowledge with regard to his/her prior knowledge.

In order to reduce the number of rules, three filters integrate the framework: operators applied over rule schemas, minimum improvement constraint filter, and item-relatedness filter. Minimum improvement constraint filter (MICF) selects only those rules whose confidence is greater with minimum than the confidence of any of its simplification s. The item-relatedness filter (IRF) Starting from the idea that the discovered rules are generally obvious, they introduced the idea of relatedness between items measuring their semantic distance in item taxonomies. This measure computes the relatedness of all the couples of rule items. We can notice that we can compute the relatedness for the items of the condition or/and the consequent, or between the condition and the consequent of the rule.

VI CONCLUSION

This paper proposes a new interactive approach to prune and filter discovered rules. First, we propose to integrate user knowledge in the processing task. Second, we propose to use the Apriori algorithm along with Rule Schema formalism extending the specifications to obtain association rules from knowledge base. We use the high confidence value based classifier for classifying the given text document to that particular domain. In proposed system, Rule Schemas bring the expressiveness of association rules by combining item constraints. The rule schema filter is based on operators applied over rule schemas allowing the user to perform several actions over the discovered rules. We propose two important operators: pruning and filtering operators. The filtering operator is composed of three different operators: conforming, unexpectedness, and exception. Proposed system scales linearly with the number of transactions. In addition, the execution time decreases a little as the number of items in the database increases. As the average transaction size increases (while keeping the database size constant), the execution time increases only gradually. We believe that the results of this approach will help decision maker for making important decisions.

VII REFERENCES

[1] Ping Chen, Rakesh Verma, Janet C. Meininger, Herman Pressler Dr.Houston, ―Semantic Analysis of Association Rules‖, Association for the Advancement of Artificial Intelligence.

[2] Claudia Marinica, Fabrice Guillet, ―Knowledge-Based Interactive Postmining of Association Rules Using Ontologies‖, IEEE Transactions On Knowledge And Data Engineering, vol. 22, no. 6, pg 784-797, June 2010 .

[3] Alaa Al Deen, Mustafa Nofal, Sulieman Bani-Ahmad, ―Classification Based On Association-Rule Mining Techniques: A General Survey And Empirical Comparative Evaluation.

[4] Faten Kharbat, Haya Ghalayini, ―New Algorithm for Building Ontology from Existing Rules: A Case Study‖, International Conference on Information Management and Engineering, pg no.12-16, 2009.

[5] Hongyu Zhang, Yuan-Fang Li, Hee Beng Kuan Tan, ―Measuring design complexity of semantic web ontologies‖, The Journal of Systems and Software 83 (2010) 803–814 at ElSEVIER.

[6] M.Z. Ashrafi, D. Taniar, and K. Smith, "Redundant Association Rules Reduction Techniques," AI 2005: Advances in Artificial Intelligence – Proc 18th Australian Joint Conf. Artificial Intelligence, pp. 254-263, 2005.

[7] R.T. Ng, L.V.S. Lakshmanan, J. Han, and A. Pang, "Exploratory Mining and Pruning Optimizations of Constrained Associations Rules," Proc. ACM SIGMOD Int'l Conf. Management of Data, vol. 27, pp. 13-24, 1998.

[8] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal, "Efficient Mining of Association Rules Using Closed Itemset Lattices," Information Systems, vol. 24, pp. 25-46, 1999.

About Author

I am Potu Narayana, Completed M.Tech from University of Hyderabad, and working as Assistant Professor in MADHIRA INSTITUTE OF TECHNOLOGY& SCIENCES, KODADA, NALGONDA, ANDHRA PRADESH. I have 4 years of experience in various programming languages developing research projects. My interest in research in data mining.