Impact Factor (SJIF): 3.632

International Journal of Advance Research in Engineering, Science & Technology

e-ISSN: 2393-9877, p-ISSN: 2394-2444

Volume 3, Issue 5, MAY-2016

A survey on Incremental association rule mining based on associative constraints

Author Hardik Dandvate[1]

[1] *Computer engineering department, Parul institute of engineering & technology, Limda, Gujarat, India*

Abstract--In the field of data mining, association is one of the biggest sub area which finds the relation that are closely associated with each other and it ruled over some basic rule mining constraints including min_sup and confidence. These two constraints are mostly used in all of the algorithms for join and prune step but for those dataset the join and prune steps becomes complicated to find the association rules, we have to consider the probability based approximation for finding the actual frequent item and generates the strong association rules. In this paper the detail survey focus on the Lift and conviction which are also the constraints of association rule mining.

Keywords- ***Association rules, support, confidence, Lift, conviction, Incremental algorithm.***

I. INTRODUCTION

In association rule mining, finding the frequent item set is one of the important task. This frequent items are generated threw different algorithms with their properties. Basically there are so many types of frequent item sets generated, which includes close frequent item set, expected frequent item set, and maximum frequent item set etc. Sometimes it is difficult to find the frequent patterns of candidates using Min_sup [8] and confidence [9].

Association rule mining [7] uses basically two constraints named min_sup and confidence but for those algorithms in which the most of the probability based approximation are required so we can't use min_sup every time. Technically Min_sup that is chosen from user. Once we fixed it we have to use it at the end of simplification of that particular data set. So for better approximation we can use the Lift and conviction constraints in association rule mining.

1.1 Lift

Lift of a rule defined as below:

Lift (A=>B) = [Support (A ∪ B)] / support A × B

It is the ratio of the observed support to that expected if X and Y were independent [2].

For example, the rule {Butter, bread} => {Milk} has a lift of 0.3\0.5×0.5=1.20

If some rule had a lift of 1, it would imply that the possibility of incidence of the antecedent and that of the respective are independent of each other. When two events are independent of each other, no rule can be generate adding those two events.

If the lift is > 1, that lets know the degree to which those two occurrences are dependent on one another, and makes those rules potentially useful for predicting the consequent in future data sets.

The value of lift is that it considers both the confidence of the rule and the overall data set.[3]

1.2 Conviction

The conviction of a rule is defined as below:

Conviction (A=>B) = 1 – support (B) / 1- confidence (A=>B)

It can be interpreted as the ratio of the expected frequency that A occurs without B (that is to say, the frequency that the rule makes an incorrect assumption) if A and B were independent divided by the observed frequency of incorrect assumption [1]. In below example, the conviction value of 1.16 shows that the rule {Butter, bread} => {milk}.

i.e., the rule {Butter, bread} => {Milk} has a conviction of [1- 0.3/ 1- 0.4] = 1.16 would be incorrect 20% more often (1.16 times as often) if the association between A and B was clearly approximated chance.

With the progress of the rapid technology of information and the need for finding useful information from dataset, data mining [4] [5] and its techniques are appeared to achieve the target of most of the frequent patterns. Support and confidence are applied on given dataset so with the help of join step and prune step it's easy to get common item sets and remove the non-essential item sets but in today's scenario there are some drawbacks and limitations in rule mining methods to find hidden item sets, close frequent item set, maximum item set etc.

The detail study makes more attention to develop the specific algorithms for close frequent item sets, maximum frequent item sets, and expected frequent item sets.

II. ASSOCIATION RULES

In general association rule [6] is defined as A=>B, where a And b are two different items of transaction database. Association rule mining generally used in a two-step process:

1). Discovering all frequent item sets whose support count Is greater than or equal to minimum support.

2). Create strong association rules from frequent Item sets, that satisfy minimum Support and minimum confidence.

The given rules are mostly used in association, but as constraints parameters Lift and conviction are not used at large stage.so it's a biggest challenging task in data mining to generate more strong association rules for Lift and conviction.

Technically there are so many algorithms are available at this stage for probability based rule mining algorithms. But Lots of hidden patterns, expected probable patterns are still remaining for extraction. Here the given survey mainly focus on the probability based algorithms [11] which are based on approximation.

2.1 Incremental association rule mining algorithm

Association rule mining uses different types of datasets and constraints to produce different types of applications. However if the new transaction are added to the database for the application of customer's purchase pattern information , means that dataset is known as incremental and frequent item sets and association rules may change. Some of the new patterns may become frequent, while some previously existing frequent patterns may become infrequent. The given figure describes the parts of incremental association rule mining as below:-

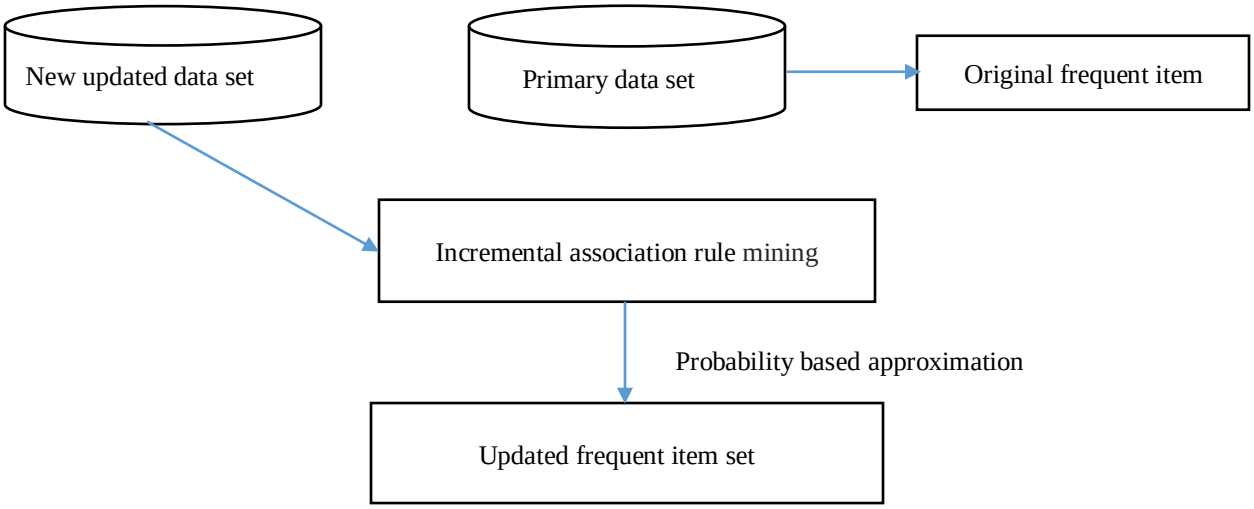

Fig 1 Incremental Association Rule Mining

Hence the given approach needs to use some strong association rules. So the given survey is suitable for developing some strategies on Lift and conviction. Here the few other incremental association rule mining algorithms [12] are described as below:-

2.2 Hash based approach for probability based incremental association rule mining [10]

The idea of hashing varies from old Apriori. It allocates the transaction entries across the hash table of buckets. It uses different hash functions for arranging buckets. Whenever the new transaction are added in database, it is little bit different then primary database, because when the new transaction are added it requires some iteration we have to perform scanning process and pruning strategies for getting the actual frequent item set fk, but in some cases after the first pass of scanning applied, join and prune step checks the common non-essential item sets in accordance with minimum support. Here the support value fixed so here the item sets whose minimum support is less then minimum support are eliminated. If this strategy applied on large dataset and if we add new transaction in it then scanning and iteration of algorithm will increase automatically. Ultimately it consumes lots of time. So here probability based strategy may be applied for further newly added transaction.

2.3 Fast update (FUP)

Fast update algorithm generally adds the new transaction in database [13]. This algorithm also uses Apriori property. When algorithm scans the database it finds the occurrences of each 1-item set. It scans the updated part of dataset first and after that it rescans the database and generates the candidate 2- item sets. The process is continued until all the frequent item sets have been discovered.

2.4 FUP 2

FUP2 is an upgraded algorithm in incremental association rule mining which varies from FUP [14]. It is basically used for both updating the item sets as well as deleting the item sets. It also trims the dataset so it is preferable algorithm as a performance compare to FUP.

2.5 EIRM

Efficient incremental association rule mining is the fastest approach for finding frequent pattern generation [15]. It is a better technique as a performance parameter and it sorts the Transaction Id according to item sets and exactly finds the frequent patterns with the use of min_sup and reduce the required number of scanning iterations to a database. The dataset requires only one scan that is the main advantage of this algorithm.

2.6 UWEP [15]

Update with Early Pruning works for large item sets transaction with early pruning when new data updated in database. It follows the influences of FUP and partition algorithm. It appoints a dynamic look-ahead strategy in updating existing large item sets by trimming and pruning superset of large item sets in newly added database. It generates small candidate item sets. The algorithm works on incremental database as well as subsidence data set. It scans the main database as well as updated database only once.

2.7 CATS tree

Compressed and transaction sequence tree approach [16] extends the Fp- growth algorithm to improve the storage compression. It generates frequent pattern without generation of candidate item sets. All the item sets are arranged in descending order. So, during the overall process, the FELINE algorithm's approach needs to cross both upwards and downwards to include frequent items. The main disadvantage of this algorithm is its cost, due to exchanging and merging of nodes to make it compact tree structure.

2.8 CAN tree

In canonical order tree [17] item sets are arranged in some accordance with canonical order. All the items can be consistently arranged in alphabetical order. This items can be defined during mining run time process. Once the occurrence of items are found then items will follow this occurrence in CAN tree for new updated database even the frequency of updated database and original database are different. CAN tree can easily arrange the nodes of the tree

without any complication and finds the combined path. It also reduce the computation time as well as tree balancing time.

III. COMPARISON OF REFERRED ALGORITHMS

Name of algorithm	method used	Performance
Hash based method	Hash function for bucket count	Excellent
Fup	Traditional Apriori	poor
Fup2	Candidate item set isolation	Average
EIRM	Pruning & trimming	Good
UWEP	Look ahead stretegy	Average
CATS tree	FELINE APPROACH	Good
CAN tree	Canonical approach	Good

CONCLUSION

Whenever the new updated data are added in database it require number of rescanning iterations. So multiple passes are required still item sets are not generated, which consumes lots of time and memory consumption. If the new items are added and common items may not generated then there is necessary requirement to Trimmed the algorithm. So the algorithm with trimming strategy should be developed and generates more powerful incremental association rules for association constraints. New strong incremental association rules should be developed based on better performance.

REFERENCES

[1] For lift reference Hahsler, Michael (2005). "Introduction to arules – A computational environment for mining association rules and frequent item sets"(PDF). Journal of Statistical Software.

[2] Web site of lift -https://en.wikipedia.org/wiki/Association_rule_learning#Lift

[3] Mining Generalized Association Rule Ramakrishnan Srikant* Rakesh AgrawalIBM Almaden Research Center San Jose, CA 95120 {srikant,ragrawal}@almaden.ibm.com

[4] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, “From data mining to knowledge discovery in databases,” AI magazine, vol. 17, no. 3, p. 37, 1996.

[5] Clifton, Christopher (2010). "Encyclopædia Britannica: Definition of Data Mining". Retrieved 2010-12-09.

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

[7] Agrawal, R.; Imieliński, T.; Swami, A. (1993). "Mining association rules between sets of items in large databases". Proceedings of the 1993 ACM SIGMOD international conference on Management of data - SIGMOD '93. p. 207.doi:10.1145/170035.170072.ISBN 0897915925.

[8] Hahsler, Michael (2005). "Introduction to arules – A computational environment for mining association rules and frequent item sets"(PDF). Journal of Statistical Software.

[9] Michael Hahsler (2015). A Probabilistic Comparison of Commonly Used Interest Measures for Association Rules. http://michael.hahsler.net/research/association_rules/measures.html

[10] Ms. Anju k.kakkad1, Ms. Anita Zal, "Incremental Association Rule Mining by Modified Approach of Promising Frequent Itemset Algorithm Based on Bucket Sort Approach", International Journal of Advanced Research in Computer and Communication EngineeringVol. 2, Issue 11, November 2013.

[11] R. Amornchewin and W. Kreesuradej, "Probability-based incremental association rule discovery algorithm"; The 2008 International Symposium on Computer Science and its Applications (CSA-08), Australia (2008).

[12] R. Amornchewin and W. Kreesuradej, "Mining Dynamic Databases using Probability-Based Incremental Association Rule Discovery Algorithm", Journal of Universal Computer Science, vol. 15, no. 12, 2009, pp. 2409-2428.

[13] D. Cheung, J. Han, V. Ng, and C. Y. Wong. Large Databases: An Incremental Updating Technique. Proceedings of the 12th International Conference on Data Engineering, pp.106—114, February 1996.

[14] D. Cheung, S. D. Lee, and B. Kao, "A General Incremental Technique for Updating Discovered Association Rules", Proceedings of the Fifth International Conference on Database Systems for Advanced Applications, pp. 185—194, April 1997.

[15] R. Feldman, Y. Aumann, and O. Lipshtat, "Borders: An efficient algorithm for association generation in dynamic databases",Journal, Intelligent Information System, 1990, pp. 61-73.

[16] W. Cheung and O. R. Zaiane, "Incremental Mining of Frequent Patterns without Candidate Generation or Support Constraint", Proceedings of the 7th International Database Engineering and Application Symposium, July 2003.

[17] C. K. Leung, Q. I. Khan and T. Hoque, "CanTree: A Tree Structure for Efficient Incremental Mining of Frequent Patterns", Proceedings of the Fifth IEEE International Conference on Data Mining (ICDM"05), 2005.