IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 16, Issue 4, Ver. VII (Jul – Aug. 2014), PP 31-34
www.iosrjournals.org

A Survey on Approaches for Mining Frequent Itemsets

S. Neelima[1], N. Satyanarayana[2] and P. Krishna Murthy[3]
[1]*Department of CSE, Research Scholar, JNTUH, Hyderabad, India.*
[2] *Department of CSE, Nagole Institute of Technology and Science, Hyderabad, India.*
[3]*Principal, Swarna Bharathi Institute of Science and Technology, Khammam, AP, India*

Abstract: *Data mining is gaining importance due to huge amount of data available. Retrieving information from the warehouse is not only tedious but also difficult in some cases. The most important usage of data mining is customer segmentation in marketing, shopping cart analyzes, management of customer relationship, campaign management, Web usage mining, text mining, player tracking and so on. In data mining, association rule mining is one of the important techniques for discovering meaningful patterns from large collection of data. Discovering frequent itemsets play an important role in mining association rules, sequence rules, web log mining and many other interesting patterns among complex data. This paper presents a literature review on different techniques for mining frequent itemsets.*

Keywords: *Association rule, Data mining, Frequent itemsets*

I. Introduction

Data mining is the discovery of hidden information found in databases and can be viewed as a step in the knowledge discovery process. Data mining functions include clustering, classification, prediction, and link analysis (associations). One of the most important data mining applications is that of mining association rules. Association rules are first introduced by Agarwal. Association rules are helpful for analyzing customer behavior in retail trade, banking system etc. Association rule can be defined as {X, Y} => {Z}. In retail stores if customer buys X, Y he is likely to by Z. Concept of association rule today used in many application areas like intrusion detection, biometrics, production planning etc. Association rule mining is defined as to find out association rules that satisfy the predefined minimum support and confidence from a given data base. If an item set is said to be frequent, that item set supports the minimum support and confidence. The problem of finding the association rules can be divided into two parts:

1. Find all frequent item sets: Frequent item sets will occur at least as frequently as a pre-determined minimum support count i.e. they must satisfy the minimum support.
2. Generate strong association rules from the frequent item sets: These rules must satisfy minimum support and minimum confidence values.

Frequent pattern mining is the process of mining data in a set of items or some patterns from a large database. The resulted frequent set data supports the minimum support threshold. A frequent pattern is a pattern that occurs frequently in a dataset. A frequent item set should appear in all the transaction of that data base.

II. Frequent Data Itemset Mining

Frequent patterns, such as frequent itemsets, substructures, sequences term-sets, phrasesets, and sub graphs, generally exist in real-world databases. Identifying frequent itemsets is one of the most important issues faced by the knowledge discovery and data mining community. Frequent itemset mining plays an important role in several data mining fields as association rules ,warehousing , correlations, clustering of high-dimensional biological data, and classification.

The frequent itemset mining is motivated by problems such as market basket analysis. A tuple in a market basket database is a set of items purchased by customer in a transaction. An association rule mined from market basket database states that if some items are purchased in transaction, then it is likely that some other items are purchased as well.

As frequent data itemsets mining are very important in mining the association rules. Therefore there are various techniques proposed for generating frequent itemsets so that association rules are mined efficiently. There are number of algorithms used to mine frequent itemsets. The most important algorithms are briefly explained here. The algorithms vary in the generation of candidate itemsets and support count. The approaches of generating frequent itemsets are divided into basic three techniques:

a. Horizontal layout based data mining techniques.
b. Vertical layout based data mining techniques.
c. Projected database based data mining techniques.

III. Algorithms for Mining from Horizontal Layout Database

In this each row of database represents a transaction which has a transaction identifier (TID), followed by a set of items.

1.1 Apriori Algorithm

Apriori algorithm is, the most classical and important algorithm for mining frequent itemsets. Apriori is used to find all frequent itemsets in a given database DB. apriori algorithm is given by Agrawal. The apriori algorithm uses the apriori principle, which says that the item set I containing item set X is never large if item set X is not large or all the non empty subset of frequent item set must be frequent also. Based on this principle, the apriori algorithm generates a set of candidate item sets whose lengths are (k+1) from the large k item sets and prune those candidates, which does not contain large subset. Then, for the rest candidates, only those candidates that satisfy the minimum support threshold (decided previously by the user) are taken to be large (k+1)-item sets. The apriori generate item sets by using only the large item sets found in the previous pass, without considering the transactions. Steps involved are:

1. Generate the candidate 1-itemsets (C1) and write their support counts during the first scan.
2. Find the large 1-itemsets (L1) from C1 by eliminating all those candidates which does not satisfy the support criteria.
3. Join the L1 to form C2 and use apriori principle and repeat until no frequent itemset is found.

1.2 Direct Hashing and Pruning (DHP) Algorithm:

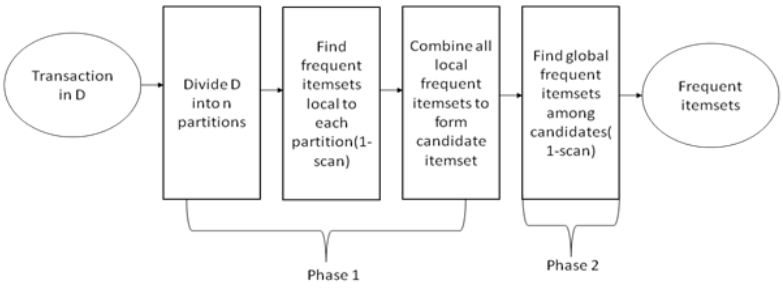
DHP can be derived from apriori by introducing additional control. To this purposes DHP makes use of an additional hash table that aims at limiting the generation of candidates in set as much as possible.DHP also progressively trims the database by discarding attributes in transaction or even by discarding entire transactions when they appear to be subsequently useless.

In this method, support is counted by mapping the items from the candidate list into the buckets which is divided according to support known as Hash table structure. As the new itemset is encountered if item exist earlier then increase the bucket count else insert into new bucket. Thus in the end the bucket whose support count is less the minimum support is removed from the candidate set.

1.3 Partitioning Algorithm

Partitioning algorithm is to find the frequent elements on the basis of dividing database into n partitions. It overcomes the memory problem for large database which do not fit into main memory because small parts of database easily fit into main memory.

The algorithm executes in two phases. In the first phase, the Partition algorithm logically divides the database into a number of non-overlapping partitions. The partitions are considered one at a time and all large itemsets for that partition are generated. At the end of phase I, these large itemsets are merged to generate a set of all potential large itemsets. In phase II, the actual support for these itemsets are generated and the large itemsets are identified. The partition sizes are chosen such that each partition can be accommodated in the main memory so that the partitions are read only once in each phase.



1.4 Sampling Algorithm:

Random sampling is a method of selecting n units out of a total N, such that every one of the CnN distinct samples has an equal chance of being selected. It is just based in the idea to pick a random sample of itemset R from the database instead of whole database D. The sample is picked in such a way that whole sample is accommodated in the main memory. In this way we try to find the frequent elements for the sample only and there is chance to miss the global frequent elements in that sample therefore lower threshold support is used instead of actual minimum support to find the frequent elements local to sample.

1.5 Dynamic Itemset Counting (DIC):

This is an alternative to Apriori Itemset Generation. In this itemsets are dynamically added and deleted as transactions are read. This algorithm also used to reduce the number of database scan. It is based upon the downward disclosure property in which this adds the candidate itemsets at different point of time during the scan. It reduce the database scan for finding the frequent itemsets by just adding the new candidate at any point of time during the run time.

1.6 Continuous Association Rule Mining Algorithm (CARMA):

CARMA brings the computation of large itemsets online. Being online, CARMA shows the current association rules to the user and allows the user to change the parameters, minimum support and minimum confidence, at any transaction during the first scan of the database. It needs at most 2 database scans. CARMA generates the itemsets in the first scan and finishes counting all the itemsets in the second scan. CARMA generates the itemsets on the fly from the transactions. After reading each transaction, it first increments the counts of the itemsets which are subsets of the transaction. Then it generates new itemsets from the transaction, if all immediate subsets of the itemsets are currently potentially large with respect to the current minimum support and the part of the database that is read. For more accurate prediction of whether an itemset is potentially large, it calculates an upper bound for the count of the itemset, which is the sum of its current count and an estimate of the number of occurrences before the itemset is generated. The estimate of the number of occurrences (called maximum misses) is computed when the itemset is first generated.

IV. Algorithms for Mining from Vertical Layout Database

In vertical layout data set, each column corresponds to an item, followed by a TID list, which is the list of rows that the item appears.

1.7 Eclat algorithm:

Equivalence Class Clustering and bottom up Lattice Traversal is known as ECLAT algorithm. This algorithm is also used to perform item set mining. It uses TID set intersection that is transaction id intersection to compute the support of a candidate item set for avoiding the generation of subsets that does not exist in the prefix tree. For each item store a list of transaction id. In this type of algorithm, for each frequent itemset i new database is created Di. This can be done by finding 'j' which is frequent corresponding to 'i' together as a set then j is also added to the created database i.e. each frequent item is added to the output set. It uses the join step like the Apriori only for generating the candidate sets but as the items are arranged in ascending order of their support thus less amount of intersection is needed between the sets.

V. Algorithms for Mining from Projected Layout Based Database

This type of database uses divide and conquer strategy to mine itemsets therefore it counts the support more efficiently then Apriori based algorithms. Tree projected layout based approaches use tree structure to store and mines the itemsets. The projected based layout contains the record id separated by column then record. Tree Projection algorithms based upon two kinds of ordering breadth-first and depth-first.

1.8 FP-Growth Algorithm:

The algorithm does not subscribe to the generate-and-test paradigms of Aprori. Instead, it encodes the data set using a compact data structure called FP-tree and extracts frequent itemsets directly from this structure. This algorithm is based upon the recursively divide and conquers strategy; first the set of frequent 1-itemset and their counts are discovered. Starting from each frequent pattern, construct the conditional pattern base, then its conditional FP-tree is constructed (which is a prefix tree.). Until the resulting FP-tree is empty, or contains only one single path. (Single path will generate all the combinations of its sub-paths, each of which is a frequent pattern). The items in each transaction are processed in L order. (i.e. items in the set were sorted based on their frequencies in the descending order to form a list).

1.9 H-mine Algorithm:

A memory-based, efficient pattern-growth algorithm, H-mine (Mem), is for mining frequent patterns for the datasets that can fit in (main) memory. A simple, memory-based hyper-structure, H-struct, is designed for fast mining. H-mine (Mem) has polynomial space complexity and is thus more space efficient than pattern-growth methods like FP-growth and tree projection when mining sparse datasets, and also more efficient than apriori-based methods which generate a large number of candidates. H-mine has very limited and exactly predictable space overhead and is faster than memory-based apriori and FP-growth. H-mine uses a H-struct new data structure for mining purpose known as hyperlinked structure. It is used upon the dynamic adjustment of

pointers which helps to maintain the processed projected tree in main memory. Therefore, H-mine proposed for frequent pattern data mining for datasets that can fit into main memory.

VI. Conclusion

The overall goal of the data mining process is to extract information from a data set and transform it into an understandable structure for further use. Association rules prove to be the most effective technique for frequent pattern matching over a decade. This paper gives a brief survey on different approaches for mining frequent itemsets using association rules. The performance of algorithms reviewed in this paper depends on support level, nature and size of the datasets.

Acknowledgements

The authors would like to thank Anonymous Reviewers for their valuable suggestions and comments. This paper has greatly benefited from their Efforts.

References

[1] Aggaraval R; Imielinski.t; Swami.A. 1993. Mining Association Rules between Sets of Items in Large Databases. ACM SIGMOD Conference. Washington DC, USA.
[2] S.Vijayarani el.al., "Mining Frequent Item Sets over Data Streams using Éclat Algorithm" International Conference on Research Trends in Computer Technologies (ICRTCT-2013).
[3] Bharat Gupta et.al., "A Better Approach to Mine Frequent Itemsets Using APRIORI AND FP-TREE Approach" .
[4] Jian Pei , Jiawei Han , Hongjun Lu , Shojiro Nishio , Shiwei Tang , Dongqing Yang "H-Mine: Hyper-StructureMining of Frequent Patterns in Large Databases".
[5] Margaret H. Dunham, Yongqiao Xiao, Le Gruenwald, Zahid Hossain "A Survey of Association Rules".
[6] Hassan Najadat , Amani Shatnawi and Ghadeer Obiedat "A New Perfect Hashing and Pruning Algorithm for Mining Association Rule" IBIMA Publishing.
[7] S.Suriya, Dr.S.P.Shantharajah, R.Deepalakshmi" A Complete Survey on Association Rule Mining with Relevance to Different Domain" INTERNATIONAL JOURNAL OF ADVANCED SCIENTIFIC AND TECHNICAL RESEARCH, ISSN: 2249-9954.
[8] Anitha Modi , Radhika Krishnan "Mining Frequent Itemsets in Transactional Database" International Journal of Emerging Technology and Advanced Engineering (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 3, Issue 2, February 2013).
[9] Ashok Savasere, Edward Omiecinski, Shamkant Navathe "An Efficient Algorithm for Mining Association Rules in Large Databases".
[10] Pramod S, O.P. Vyas "Survey on Frequent Item set Mining Algorithms ",International Journal of Computer Applications (0975 - 8887) Volume 1 – No. 15.
[11] Shruti Aggarwal, Ranveer Kaur" Comparative Study of Various Improved Versions of Apriori Algorithm" International Journal of Engineering Trends and Technology (IJETT) - Volume4Issue4- April 2013

Mrs. S. Neelima, working as an Associate professor in the department of Computer Science and Engineering. Her research areas include Cloud Computing, Data Ware housing and data mining, Computer Networks and Network Security.

Dr.N.Satyanarayana, M.Sc, M.Phil, AMIE (ET), M.Tech (CS), Ph.D (CSE), MISTE, MCSI, received his Ph.D degree in Computer Science & Engineering from Acharya Nagarjuna University, currently working as a Professor in department of CSE at Nagole Institute of Technology & Science. His research interests include Advanced Computer Architecture, Networking, Data Mining and Wireless Communications

Prof. Pannala Krishna Murthy has secured his B.Tech from SSGMCE- Shegoan. He obtained his Masters degree from JNTU College of Engineering, Kukatpally, Hyderabad. He received Doctor of Philosophy in Electrical Engineering for his work titled "Analysis & Identification of HVDC system faults using Wavelet Transforms" from JNTU, Hyderabad on in the year 2010. He has been working Professor & Principal of SBIT. He presented and published internationally over 26 Technical papers. He guided more than 15 PG projects and guiding 6 PhD Scholars.