International Journal For Technological Research In Engineering
Volume 3, Issue 7, March-2016 *ISSN (Online): 2347 - 4718*

CONSTRAINED BASED CORRELATED ITEM SET MINING

Nirav Gajjar[1], Naimish R Patel[2]
[1]PG Scholar, [2]Assistant Professor, Computer Department, SCET, Saij, Kalol, Gujarat, India

ABSTRACT: Extracting frequent item sets is an important task in many data mining applications. When data are very large, it becomes mandatory to perform the mining task by using an external memory algorithm. Certain limitations of the partitioning techniques adopted by external memory algorithms for extracting all the frequent item sets, when applied to closed item sets mining. We introduce the first algorithm for mining closed item sets out of core. The algorithm exploits a divide-et-imperia approach, where the input dataset is split into smaller partitions, such that not only they can be loaded, but also they can be mined entirely into the main memory.

I. INTRODUCTION

Ever since the introduction of association rules, researchers have studied various problems related to mining interesting patterns from large databases [1]. The importance of data mining has been increased rapidly for business domains like marketing, financing and telecommunications. In recent decade the development of economic is violent and swift. Information enhances unceasingly in a highest level. So the organizations and agencies have collected the massive business data. The business organizations urgent need to discover the valuable information and knowledge from the magnanimous data. The typical example of mining a frequent item set is market basket analysis through discovering the interactions between the different merchandise that the customer puts in "the basket"[1]. Data mining algorithms are helpful in digging out hidden previously unknown information from existing data [2] Positive and negative association rule:

1.1 Efficient mining of Positive and Negative Association Rules with weighted FP - Growth [1][6]

Analyzing data from different perspectives and summarizing it into useful information is what the process of data mining which can be used to increase revenue, cuts costs, or both. Association rule mining is a data mining technique that finds frequent patterns or associations in large data sets. Association rule mining is recognized as positive association rule mining. The association rules are an important research content in data mining which finds frequent patterns or associations in large data sets. An association rule is an implication of the form A ⇒ B, where A and B are frequent item sets in a transaction database which are called as positive association rules and A ∩ B = φ. In practical applications, the rule A ⇒ B can be used to predict that „If a occurs in a transaction, then B will likely also occur in the same transaction, and we can apply this association rule to recommend who purchase B or Efficient mining of Positive and Negative Association Rules with weighted FP - Growth 26 Placing B close to A in the stores layout, such application are expected to provide more convenience for customers, and increasing product sales. Recently much work is focused on finding alternative patterns, including unexpected patterns, which are also known as surprising patterns for example while „bird (X) ⇒ flies (X)" is the well known fact, an exceptional rule is bird (X), penguin(X) ⇒ ¬ flies(X)" which indicates negative term and can be treated as a special case of negative rules. In paper [1] extends traditional associations to include association rules of the form (A ⇒ ¬B), (¬A ⇒ B) and (¬ A ⇒ ¬B) which indicates negative associations between item sets, and are called negative association rules. Negative association rules assist in decision making which also help the companies to hunt more business chances through infrequent item sets of interests. Negative associations provide vital information to data. Association rule mining is a data mining approach that is used often in traditional databases and usually to find the positive association rules. As compared to positive associations, negative association takes over lots of search space. In turn needs more execution time as well as consumes more memory. So for faster mining process there is need of precise and special patterns with accuracy. For fast and accurate decision we need to have some most significant patterns and prioritize the selection of target item sets according to their significance in the data set. The main motivation of this project is to provide the data processing accurately and efficiently with significant or prioritized data items from large data base.

Prior association rule model assumes that items have the same significance without taking account of their attribute within a transaction or within the whole item space. The main objective of this project is to implement the better system for data processing with excellent time and space management. The goal of using weighted support is to make use of the weight in the mining process and prioritize the selection of target item sets according to their significance in the data set rather than their frequency alone.

1.2 Mining Frequent Item sets with Convertible Constraints [2][4][7]

Frequent pattern mining often generates a very large number of frequent item sets and rules, which reduces not only the efficiency but also the effectiveness of mining since users have to sift through a large number of mined rules to find useful ones. Item set constraints have been incorporated into association mining. A systematic method for the incorporation of two large classes of constraints—anti-monotone A method for mining association rules [1][6] in large, dense databases by incorporation of user-specified constraints that ensure every mined rule offers a predictive

advantage over any of its simplifications, is developed in Constraint-based mining of correlations, by exploration of anti monotonicity and succinctness, as well as monotonicity, is studied in.[4][7]

As a general picture, constraints (only involving aggregate functions) can be classified into the following categories according to their interactions with the frequent item set mining process: anti-monotone, monotone, succinct and convertible, which in turn can be subdivided into convert ibleanti-monotone and convertible monotone. The inter section of the last two categories is precisely the class of strongly convertible constraints (which can be treated either as convertible anti-monotone or monotone by ordering the items properly).

- Mining frequent item sets with convertible constraints.
- Mining frequent item set with Monotone constraints.
- Mining frequent item set with Anti-monotone constraint

1.3 An Efficient Algorithm for Frequent Closed Item sets Mining [3]

There are basically two types of algorithms to mine frequent item sets, breadth-first algorithms and depth-first algorithms. The breadth-first algorithms, such as A priori and its variants, apply a bottom-up level-wise search in the item set lattice. On the other hand, depth-first algorithms such as FP-growth search the lattice bottom-up in depth-first way.

II. FP-TREE AND FP-GROWTH METHOD

The FP-tree is a compact data structure for storing all necessary information about frequent item sets in a database. Every branch of the FP-tree represents a frequent item set, and the nodes along the branch are ordered decreasingly by the frequency of the corresponding item, with leaves representing the least frequent items. Each node in the FP-tree has three fields: item-name, count and node-link, when item-name registers which item this node represents, count registers the number of transactions represented by the portion for the path reaching this node, and node-link links to the next node in the FP-tree carrying the same item-name, or null if there is none. The FP-tree has a header table associated with it. Single items are stored in the header table in decreasing order of frequency. Each entry in the header table consists of two fields, item-name and head of node-link (a pointer pointing to the first node in the FP-tree carrying the item-name). Compared with A priori and its variants which need several database scans, the FP-growth method only needs two database scans when mining all frequent item sets.[10] In the first scan, all frequent items are found. The second scan constructs the first FP-tree which contains all frequency information of the original dataset. Mining the database then becomes mining the FP-tree.

III. CFI-TREE

In algorithm FP close introduces the Closed Frequent Item set tree (CFI-tree) as the special data structure to store CFI. The CFI tree ensembles an FP-tree. It has a root labeled with "root". Children of the root are item prefix sub trees. Each node in the sub tree has four fields: item-name, count, node-level and node-link. All nodes with same item-name are linked together. The node-link points to the next node with same item-name.

A header table is constructed for items in the CFI-tree; the item order in the table is same as the item order in the first FP-tree constructed from the first scan of the database. Each entry in the header table consists of two fields: item-name and head of a node-link. The node-link points to the first node with the same item-name in the CFI-tree. Following is how CFI-tree can be constructing.

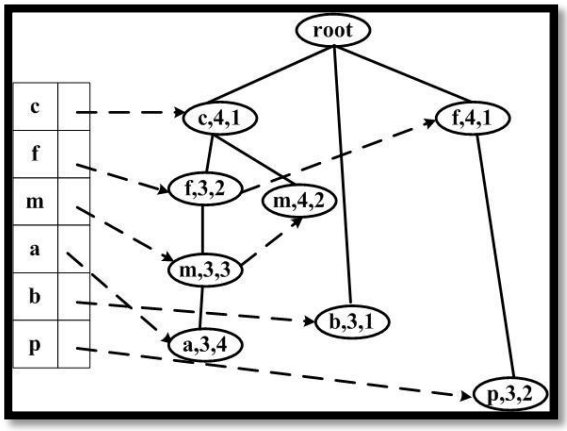

Fig-CFI Tree model

IV. FCFIA: MINING CFI

We extend the FP-growth method and get algorithm FCFIA described in Pro Like FP-growth, algorithm FCFIA is also recursive. In the initial call, an FP-tree is constructed from the first scan of the database. A linked list head contains the items that form the conditional pattern base of the current call. If there is only one single path in the FP-tree, every frequent item set X generated from this single path together with head is an frequent item set, then we check if item set head ∪ X is a frequent closed item set.

Constraint Programming for Correlated Item set Mining [5][9]

An essential step in building such classifiers is to discover rules that predict the target attribute well. A common approach in the machine learning community is to learn one such rule by applying a heuristic in a greedy algorithm. In the data mining community, on the other hand, it has been studied how to find such rules under constraints in an optimal way. The traditional example is the search for all association rules [1]. Usually however many association rules can be found and their direct application for classification is impossible. To focus the discovery of rules more towards classification, the use of correlation constraints has been studied [8][11], leading to algorithms for correlated or discriminative item set mining. Assumed given is a function f which scores every item set based on how well it correlates with a target attribute. Examples of such functions are _2 and information gain. The problem is to find the k item sets that score highest with respect to f. For k = 1, this problem can be thought of as finding the optimal rule under function f, instead of an arbitrary good one, as common in

machine learning. This problem is known to be NP-complete, and hence, a general, efficient algorithm cannot be expected to exist.

Main principles in constraint programming are:

- Problems are specified declaratively by providing constraints on variables within domains;
- Solvers find solutions by constraint propagation.

Inverse-free Berlekamp–Massey Sakata Algorithm and Small Decoders for Algebraic Geometric Codes

In fast decoding of codes, Berlekamp–Massey–Sakata (BMS)[5]algorithm is often used for finding the location of errors, and the evaluation of error-values is done by using outputs of BMS algorithm. Reed–Solomon codes have the features of high error-correcting capability and less complexity for the implementation of encoder and decoder.

On the other hand, codes on algebraic curves have the issues related to the size of decoders as well as the operating speed of decoders. In particular, we notice that RS coded encoders need no inverse-calculator of the finite field (no finite-field inverter). The extended Euclidean algorithm for RS codes has no divisions, and this enables us to operate compactly and quickly in calculating error-locator and error-evaluator polynomials. One cannot execute the determination of unknown syndromes because of breaking the generation of candidate values of unknown syndromes for majority voting. Unfortunately, the elimination of finite-field divisions seemed to be a difficult problem in this regard.

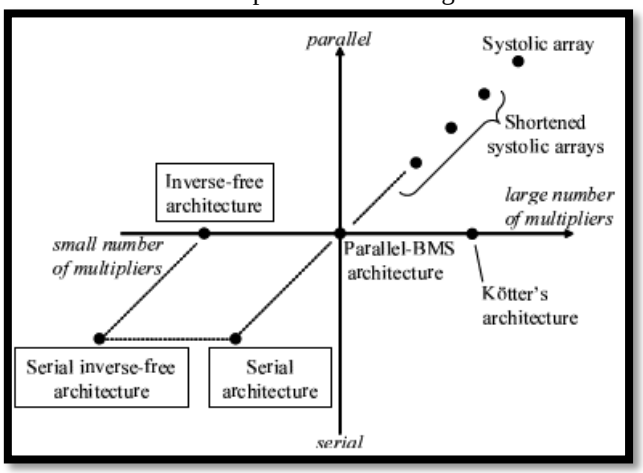

Figure - [Error Location mechanism]

In this research, we effectively overcome this difficulty. Namely, we decode such codes with the only known syndrome values from received code-words. So far the type and amount of errors that could be corrected if one does not determine unknown syndromes have not been clear.

V. PROPOSED METHODOLOGY

Introduction

This proposed algorithm is very useful to find frequent item set among large dataset quickly and with less external memory. Limitation of a priory algorithm is that, it is slow and bottle neck to generate candidate key, and also requires many database scan.

Our proposed algorithm is solution to improve overall performance to mine frequent item set from given set of data with different constrains. Some of algorithms which are useful in Berlekamp–Massey–Sakata (BMS)[5] algorithm is often used for finding the location of errors, and the evaluation of error-values is done by using outputs of BMS algorithm.

We have adopted a top-down transposition based searching strategy and a new support counting method

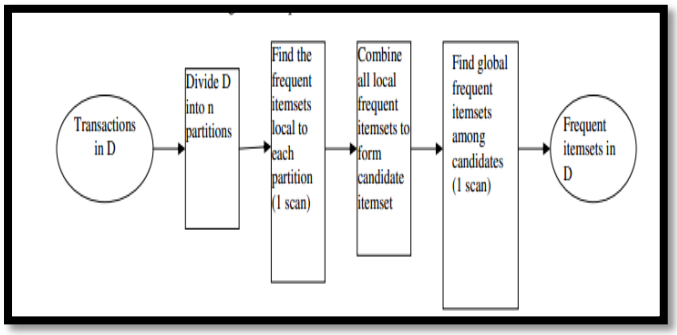

Fig-Proposed Method

D be the database corresponding to i^{th} branch of the organization, I = 1, 2 ….n. Patterns in multiple databases could be grouped into the following categories based on the number of databases. Local patterns, global patterns and patterns which are not global or local. A local pattern is based on a branch database. A global pattern is common in the entire database taken under study. Dynamic item set counting (adding candidate item set different points during a scan): A dynamic item set counting technique was proposed in which the database is partitioned into blocks marked by start points. It is created by lattice technique using DIC algorithm

DIC algorithm

- Mark the empty item set with a solid square. Mark all the 1-itemsets with dashed circles. Leave all other item sets unmarked.
- While any dashed item sets remain:
    - Read M transactions (if we reach the end of the transaction file, continue from the beginning). For each transaction, increment the respective counters for the item sets that appear in the transaction and are marked with dashes.
    - If a dashed circle's count exceeds min supp, turn it into a dashed square. If any immediate superset of it has all of its subsets as solid or dashed squares, add a new counter for it and make it a dashed circle.

VI. CONCLUSION

We have presented a novel algorithm able to mine all the frequent closed item sets from a transactional database using a limited amount of main memory. To our best knowledge, this is the first external memory algorithm for mining closed

item sets. The two main contributions of this paper are, on the one hand, the optimization of an already known projected based partitioning technique adapted to our framework, and, on the other hand, an innovative merging technique of the local results extracted from each partition. we have reduced the problem of merging partial solutions to an external memory sorting problem.

REFERENCES

[1] Ms. Varsha N Kavi, Mr.Divyesh Joshi, "Efficient mining of Positive and Negative Association Rules with weighted FP – Growth" Volume 10, Issue 5 (May 2014), PP.25-31
[2] Jian Pei, Jiawei Han, Laks V.S. Lakshmanan, "Mining Frequent Itemsets with Convertible Constraints".
[3] Lisheng Ma, Yi Qi, "An Efficient Algorithm for Frequent Closed Itemsets Mining" , 2013 International Conference on Computer Science and Software Engineering
[4] Siegfried Nijssen, Tias Guns, Luc De Raedt, "Constraint Programming for Correlated Itemset Mining", DepartementComputerwetenschappen K.U. Leuven Celestijnenlaan 200A, 3000 Leuven, Belgium.
[5] Hajime Matsui, Seiichi Mita ,"Inverse-free Berlekamp–Massey–Sakata Algorithm and Small Decoders for Algebraic-Geometric Codes"
[6] R. Agrawal et al. Mining association rules between sets of items in large databases. SIGMOD 1993.
[7] Siegfried Nijssen, Tias Guns, Luc De Raedt "Constraint Programming for Correlated Itemset Mining" ,DepartementComputerwetenschappen K.U. Leuven Celestijnenlaan 200A, 3000 Leuven, Belgium.
[8] HuiXiong, ShashiShekhar, P. N. Tan, and Vipin Kumar, "Exploiting a support-based upper bound of Pearson's correlation coefficient for efficiently identifying strongly correlated pairs,"KDD'04, pp. 334–343, August 22-25, 2004, USA.
[9] R. J. Bayardo, R. Agrawal, and D. Gunopulos.Constraintbasedrule mining on large, dense data sets. In Proc. 1999Int. Conf. Data Engineering (ICDE'99), Sydney, Australia, Apr. 1999.
[10] SametCokpinar, TaflanImreGunden, "Positive and negative rule mining on XML data streams in database as a service concept", Expert Systems with Applications, Vol.39, pp.7503-7511, 2012.
[11] G¨ostaGrahne, Laks V. S. Lakshmanan, Xiaohong Wang" Efficient Mining of Constrained Correlated Set"
[12] https://en.wikipedia.org
[13] http://www.tutorialspoint.com

