ISSN: 2321-7782 (Online)
Volume 3, Issue 11, November 2015
International Journal of Advance Research in Computer Science and Management Studies
Research Article / Survey Paper / Case Study
Available online at: www.ijarcsms.com

Frequent Pattern Mining using Parallel Architecture of Artificial Bee Colony

Sanjay Patel[1]
Computer Engineering Department,
Government Engineering College, Katpur,
Patan, Gujarat. Pin -384265, India

K. Kotecha[2]
Vice Chancellor,
Parul University, Waghodia,
Vadodara, Pin–391760, India

Abstract: ***Data mining is the process of extracting unknown patterns from large amount of data. Frequent pattern mining is one of the research areas of Data Mining. Most of the Existing Frequent Pattern Mining algorithms are used to deal with tree based approach. Very few algorithms are based on directed graph approach. Undirected graph based methods for frequent pattern mining are rare. Artificial Bee colony architecture is partially analogous to parallel processing. It is about the behavior of artificial agents, here, food exploitation done by different kinds of bees. Problem being a combinatorial optimization, there are multiple feasible solutions possible so multiple food sources are required to be evaluated simultaneously. One of the Undirected Graph based approach that has been solved previously with the serial method; try has been made to solve it with the parallel approach. Here in this Paper Architecture for frequent Pattern Mining is designed to improve the performance in terms of time and space. At the end the comparison of results with different methods is shown.***

Keywords: Artificial Bee Colony, Association Rule Mining, Data Mining, Frequent pattern Mining

I. INTRODUCTION

In this information age, it is possible to store a large amount of data cheaply in both financial sense and physical sense. Manually retrieval of useful information from huge amount of data is really a typical and time consuming task. Due to this scenario, Data Mining comes into the picture. Basically, Data mining is the process of finding unknown and potentially useful information from the bunch of data. Data Mining is also known as a Knowledge Discovery from Data. There are several steps of Knowledge Discovery from Data, which includes Data Cleaning, Data Integration, and Data selection, Data Transformation, Data Mining, Pattern Evaluation and Knowledge Presentation. Association Rule Mining is one of the techniques for data mining. Frequent Pattern Mining is a part of Association Rule Mining. Starting from AIS **[33]** and SETM, so many algorithms have been proposed for finding frequent patterns from huge amount of data. The Problem of Association Rule Mining was also introduces in **[33]**. The problem of Association Rule Mining was improved later on by Apriori Algorithm. Apriori was the first algorithm which utilizes the Candidate Generation and Test approach [27]. Several improvements over Apriori Algorithm were proposed so far as Hash based Technique, Transaction Reduction, Partitioning, Sampling and Dynamic Itemset Counting [15]. Due to limitations of Candidate Generation and Test approach, Han et al. (2000) [16] devised FP-Growth method that mines the complete set of Frequent itemsets without Candidate Generation and Test. It Uses the Divide and Conquer Strategy. It requires two scan of the datasets. Association rule mining works on the trial and error basis means initial mining results may not be satisfactory. It may be possible that user provided support threshold needs to be changed based on the results. In FP-Growth approach when this happens the whole procedure needs to be started from scratch which is time consuming procedure. With this idea in mind, Cheung W. (2002) [4] projected CATS Tree (Compressed Arranged Transaction Sequences) approach. CATS Tree works on the principle of Interactive mining, “Build once, mine many”. Swapping, merging, deletion of the node is the problem of CATS tree. It takes too much time. Also storage is the constraint for this type of tree structure. Researchers assumed unlimited amount of memory but in practical applications this is not possible. CKS Leung et al. [24] proposed a new tree

structure called Can-Tree (Canonical Tree). In comparison to CATS Tree, here all the items are ordered according to some specific ordering, for example Lexicographical or Alphabetical. Available data can be in any order, to arrange the data in some specific sequence is also a typical task. This is the additional overhead of the mechanism. The tree size is also dependent on the items appearing in the transactions.

R.S. Thakur et al. (2008) proposed a graph based algorithm. The sole feature of this algorithm is that it scans the entire database only once. Directed Graph is created during scanning the database. The algorithm overcomes drawbacks of the FP-Growth Approach. The details of the algorithm are available in literature (R. S. Thakur et al.). Based on FP-Growth the Graph based FP-Growth-Graph was proposed by Vivek Tiwari et al. in 2010. FP Growth tree uses tree for arranging the items before mining, where more than one node can contain single item which causes repetition of same item and needs more space to store many copies of same item. Page fault is also one of the causes of it. PGMiner, a novel graph based algorithm proposed by H.D.K. Moonesinghe et al. for mining frequent closed itemsets. The first step in this approach is to construct a prefix graph structure and decomposing the database to variable length bit vectors. Mining process in detail is available in literature (H.D.K. Moonesinghe et al.).

II. RELATED BACKGROUND

This section explores the work done in the area of frequent pattern mining. Most of the common methods are discussed herewith. AIS and SETM are the first of all algorithms invented for frequent pattern mining. Candidates are generated on the fly during the scanning process. Unnecessarily generation of too many candidate sets and counting is the disadvantage of this method. While in the case of SETM algorithm, the candidates are generated on the fly but counted at the end of the pass. New candidate generation procedure is same as AIS but the TID of the generated transaction is saved in sequence. (R. Agrawal and R. Srikant, R.Agrawal). Apriori Algorithm was invented by R. Agrawal and R. Srikant in 1994 for frequent pattern mining. Apriori uses the iterative approach known as a level-wise search where n itemsets are used to explore n+1 itemsets. Variations of the Apriori Algorithm are discussed herewith for the improvement in the performance (F. Geerts et al., Jong Soo Park et al. Junqiang Liu).

- Hash-based technique
- Transaction Reduction
- Partitioning
- Sampling
- Dynamic Itemset Counting
- AprioriTid
- AprioriHybrid
- Direct Hashing and Pruning
- Partition Algorithm

The following are using the Vertical data representation for frequent pattern mining.

- ECLAT (Equivalence Class Transformation) (J. Han and Micheline Kamber ,Rajanish Dass et al.)
- VIPER (Vertical Itemset Partitioning for Efficient Rule Extraction) (M. Song and S. Rajasekaran)
- MAFIA (Maximal Frequent Itemset Algorithm) (M. Song and S. Rajasekaran)

The candidate generation and test process is the major setback for Apriori-like methods. FP-growth algorithm solves this problem. Transactional database information is important for mining frequent patterns. So, if that information can be stored in compact data structure, it can help in frequent pattern mining. By this idea in mind, a compact data structure, called FP-tree has been developed by Jian Pei to store complete but no redundant information for frequent pattern mining.

Variations of the FP-growth Approach

- AFPIM (Adjusting FP-tree for Incremental Mining) (S. K. Tanbeer et al.)
- EFPIM (Extending FP-tree for Incremental. Mining) (S. K. Tanbeer et al.)
- FUFP-tree (Fast Updated Frequent Pattern Tree) (S. K. Tanbeer et al.)
- FP-Max (Maximal Frequent Itemsets) (Gosta Grahne et al.)
- FP-Max* (Maximal Frequent Itemsets – Advanced) (Gosta Grahne et al.)
- FP-Growth* (Frequent Pattern Growth – Advanced) (A.M. Said et al., Gosta Grahne et al.)
- CFP- Tree (Compresses FP-Tree) (Fei Chen et al.)
- H-Struct and H-Mine (Memory Based Hyper-Structure Mining) (J. Pei, J. Han, et al.)
- AFOPT Algorithm (A.M. Said et al.)
- NONORDFP Algorithm (A.M. Said et al.)

The CATS (Compressed Arranged Transaction Sequences) tree is an expansion of FP (Frequent Pattern) Tree and it includes all elements of FP-Tree. Still, there are few key differences between the two data structures. A CATS Tree contains all items of the transaction. Every item in the dataset has a node in the header and all of them consist of the overall frequency of the item in the dataset (Carson Kai-Sang Leung et al., Cheung W.). In CATS tree competency for incremental mining is not clear (where the database is changed frequently). CKS Leung et al. [24] proposed CanTree to overcome drawback of CATS tree. The Construction of the CanTree requires only one database scan.

List of the directed graph based frequent pattern mining methods:

- Efficient Graph Based Algorithm (EGBA) for Mining Frequent Itemsets (P. Deepa Shenoy et al.).
- Prefix Graph based frequent itemset mining with efficient Flow-Based Pruning Strategy (H.D.K.Moonesinghe et al.)
- Graph theoretic Based Algorithm for Mining Frequent Patterns (R. S. Thakur et al.)
- FP-Growth-Graph: A Graph based approach for mining Frequent Itemsets (Vivek Tiwari et al.)

III. PROPOSED WORK

The method described here is based on undirected graph. Each item is a node and items in a single transaction are linked with the edges means every item node is linked i.e. edge is made; with every other item node in a single transaction. If number of unique items (nodes) in a transaction is n then it has nC2 edges in graph. Edge has two weights (attributes) namely fid(Frequent item set Identifier) and c (Counter).Each transaction is associated with a unique prime number starting from the first prime number 2. The bottleneck in serial implementation is to deal with very large multiplication of numbers, finding greatest common divisor and factorizing the same. Empirical solution to the problem is Divide and Conquer. The Parallel architecture of proposed method is partly inspired by and analogous to Artificial Bee Colony (D. Karaboga, V. Tereshko et al.). It is about the behaviour of artificial agents. Food exploitation is done by different kind of bees. As the problem being a

combinatorial optimization, there are multiple feasible solutions possible so multiple food sources are required to be evaluated simultaneously.

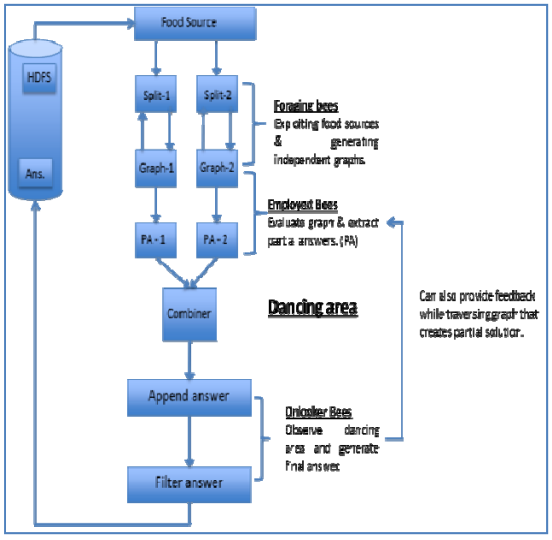

Figure 1: Parallel architecture block diagram - ABC

This design can be extended to multiple processors/nodes attached in parallel so as to gain efficiency over serial version (Mihai Cuibus et al.).

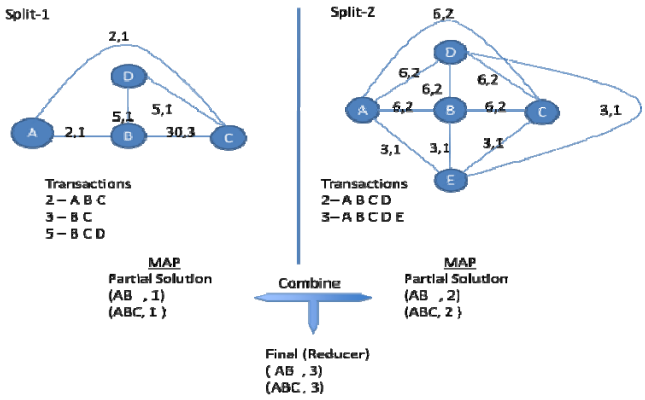

Figure 2: Block Diagram of Parallel Procedure

IV. Experimental Results

Parallel approach is the extension of the Serial approach. In Parallel Approach both the concepts are combined. In all the split, the ACO is applied individually. When the results are merged using the combiner (Reducer), the concept of ABC (Artificial Bee Colony) – Parallel Architecture is applicable here. The comparison is made for Parallel approach and Serial Approach. Figure 3 shows a comparison for Time vs. Minimum Support. Straight away the Parallel Approach performs better. Figure 4 shows a comparison for Time vs. Number of transactions.

V. Conclusion

- Input data is equally distributed across different processors
- All edges with support less than required min support is removed in serial version
- In parallel version, graph pruning is not possible
- Parallel approach optimizes the performance of ACO based frequent mining method
- The model can be supported by both Incremental and Interactive mining methods
- All implementations deal with the graph based database representations

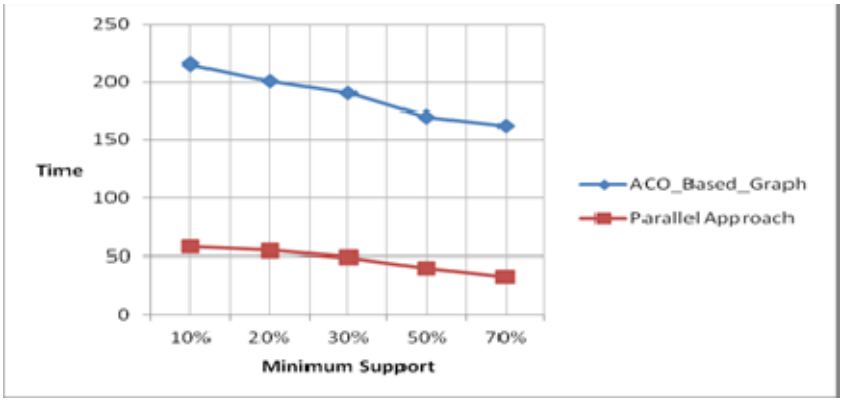

Figure 3: Time vs. Min_Sup

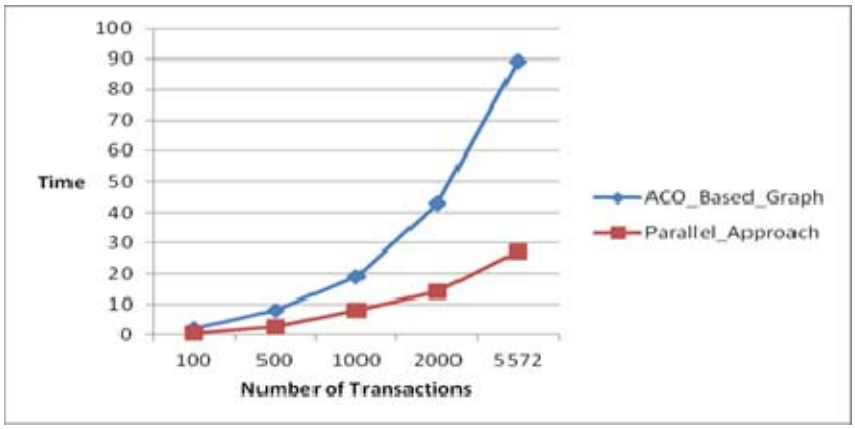

Figure 4: Time vs. Number of Transactions

Benefits over Serial implementation

» The major benefit over serial implementation is dealing with very large numbers which is reduced, that is, multiplication of large primes and further calculation of greatest common divisor and factorization to calculate count.

» Next benefit is after the reduce phase when item sets are filtered, answer for different support values (Interactive Mining) can be obtained without re-iterating the complete process unlike in Serial implementation.

» This Concept can even be extended to implement incremental mining, that is, if in future few transactions arrive, create one split (node) for separate calculation and add to the old frequency of those item sets.

Limitations

» For very large data sets it shows same problem as in serial implementation.

» i.e., Convergence to very large value of Prime id and its multiplication.

» Implementation is complex compared to serial implementation.

» In parallel, traversal of every path in the graph is done in each map task which is also for infrequent items that is not required in serial.

References

1. A.J.T. Lee, R.W. Hong, W.M. Ko, W.K. Tsao, H.H. Lin (2007), "Mining spatial association rules in image databases", Information Sciences 177 (7) (2007) 1593–1608.
2. Alva Erwin, Raj P.Gopalan and N.R.Achuthan(2007), "CTU-Mine: An efficient High Utility Itemset Mining Algorithm using Pattern growth approach" Seventh International conference on Computer and Information Technology,2007.
3. Carson Kai-Sang Leung et al. (2007), "CanTree: a canonical- order tree for incremental frequent-pattern mining", Knowledge and Information Systems, 2007, 11(3), Page 287-311.
4. Cheung W.(2002), "Frequent Pattern mining without candidate generation or support constraint." Master's thesis, University of Alberta, 2002, SPRING '03, doi.ieeecomputersociety.org/10.1109/IDEAS.2003.1214917.
5. Christian Borgelt(2005)" An Implementation of the FP-growth Algorithm" OSDM'05, August 21, 2005, Chicago, Ilinois, USA.
6. Elena Baralis et al. (2013), " P- Mine: Parallel itemset mining on large datasets", Data Engineering Workshops (ICDEW), 2013 IEEE 29th International Conference on, Date 8 – 12 April, 2013.
7. En Tzu Wang et al. (2011) , " Mining Frequent Itemsets over distributed data streams by continuously maintaining a global synopsis" Data Mining and Knowledge Discovery, Springer, September 2011, Volume 23, Issue 2, pp 252-299
8. G. Chen, Q. Wei (2002), "Fuzzy association rules and the extended mining algorithms", Information Sciences 147 (1–4) (2002) 201–228.
9. G. Piatetsky-Shapiro et al. (1991), " Knowledge Discovery in Databases" , MIT Press, 1991.
10. Geoffrey I. Webb(2010), " Self-Sufficient Itemsets: An Approach to Screening Potentially Interesting Association Between Items" , ACM Transactions on Knowledge Discovery from Data, Volume 4, No.1, Article 3, January 2010.
11. Guimei Liu et al. (2013), " A Flexible Approach to finding Representative Pattern Sets", IEEE Transactions on Knowledge and Data Engineering, 04 Feb. 2013. IEEE computer Society Digital Library. IEEE Computer Society.
12. H.D.K. Moonesinghe, S. Fodeh, P.N. Tan(2006), "Frequent Closed Itemset Mining using Prefix Graphs with an efficient Flow-Based Pruning Strategy" , Proceedings of the sixth International Conference on Data Mining (ICDM '06) IEEE Computer Society.
13. Irina Tudor(2008), "Association Rule Mining as a Data Mining Technique" , Buletinul Universitatii Petrol Gaze din Ploiesti, Seria Matematica-Informatica-Fizica , Vol. LX, No. 1/2008, page 49 – 56.
14. J. Han, G. Dong, G. Yin(1999), "Efficient mining of partial periodic patterns in time series database" , In: Proceedings of IEEE International Conference on Data Mining, 1999.
15. J. Han and Micheline Kamber, Book :"Data Mining, Concept and Techniques", 578 pages, books.google.co.in.
16. J. Han, J. Pei, Y. Yin(2000). "Mining frequent patterns without candidate generation." SIGMOD 2000.
17. J. Pei, J. Han, et al.(2001), " H-Mine: Hyper-Structure Mining of Frequent Patterns in Large Databases." ICDM 2001.
18. Jien Pei, "Pattern-Growth methods for frequent pattern mining " Ph. D. Thesis, Simon Fraser University, 2002.
19. Jilles Vreeken et al.(2011), " KRIMP : mining itemsets that compress" Data Mining and Knowledge Discovery, Springer, July 2011, Volume 23, Issue 1, pp 169-214.
20. K. Kotecha et al. (2011), "Frequent Pattern Mining Using Graph Based Approach" , International Journal of Computer Science Research and Application 2011, Vol. 01, Issue. 02
21. K. Kotecha et al. (2013), "Incremental Frequent Pattern Mining using Graph based approach" , International Journal of Computers & Technology,Volume 4 No. 2, March-April, 2013, ISSN 2277-3061
22. Liang Wang et al.(2012),"Efficient Mining of Frequent Itemsets on Large Uncertain Databases", IEEE Transactions on Knowledge and Data Engineering, Vol. 24, No. 12, December 2012.
23. Q. Li, L. Feng, A. Wong(2005), "From intra-transaction to generalized inter-transaction: landscaping multidimensional contexts in association rule mining", Information Sciences 172 (2005) 361–395.
24. Q. I. Khan, T. Hoque and C.K. Leung(2005), " CanTree : A Tree structure for Efficient Incremental mining of Frequent Patterns". Proceeding of the Fifth International conference on Data Mining (ICDM'05), csdl2.computer.org/.../&toc=comp/proceedings/icdm/2005/2278/00/ 2278toc.xml &DOI=10.1109/

    ICDM 2005.
25. P. Deepa Shenoy et al.(2003) , " Compress and Mine: An Efficient Graph Based Algorithm to Generate Frequent Itemsets" , Advance Computing and Communicating Society , 2003.
26. P. M .Kanade, et al. (2007), " Fuzzy Ants and Clustering", IEEE Transactions on Systems, Man and Cybernatics, Volume 37, No. 5, September 2007.
27. R. Agrawal, R. Srikant(1994) , "Fast algorithms for mining association rules, in: Proceedings of the 20th International Conference on Vary Large Data Bases", 1994, pp. 487–499.
28. R. J. Kuo et al. (2007) , " Association Rule mining through the ant colony system for National health Insurance Research Database in Taiwan" , An International Journal of Computers and Mathematics with applications 54, Science Direct, pp 1303-1318.
29. R. S. Parpinelli et al. (2002), " Data Mining with an Ant Colony Optimization Algorithm", IEEE Transactions on Evolutionary Computing, Volume 6, No. 4, August 2002.
30. R. S. Thakur, R.C. Jain and K.R. Pardasani(2008), " Graph Theoretic Based Algorithm for Mining Frequent Patterns" Neural Networks, IJCNN (IEEE world Congress on Computational Intelligence). 1-8 June, 2008.
31. Rajnish Dass and Ambuj Mahanti, " An efficient heuristic search for Real-Time frequent pattern mining". International Conference on System Sciences – 2006, ieeexplore.ieee.org/iel5/10548/33362/01579371.pdf
32. Rajanish Dass et al. (2006) ," An Efficient Algorithm for Real-Time Frequent Pattern Mining for Real-Time Business Intelligence Analytics", Proceedings of the 39th International Conference on System Sciences- IEEE, 2006

33. Rakesh Agrawal et al. (1993), " Mining Association Rules between set of items in large databases", In proceedings of ACM SIGMOD Conference on Management of Data (SIGMOD '93) pages 207- 216, May 1993.

34. S. Shankar et al.(2009),"Utility Sentient Frequent Itemset Mining and Association Rule Mining : A Literature Survey and Comparative Study" , International Journal of Soft Computing, ISSN: 1453 – 2277 Issue 4 (2009), pp 81-95.

35. S. Shankar et al. (2009), " A Novel Utility Sentient Approach for Mining Interesting Association Rules" , IACSIT International Journal of Engineering and Technology, Volume 1, No. 5, December 2009.

36. S. K. Tanbeer et al. (2008), " Efficient single – pass frequent pattern mining using a prefix tree" , Information Sciences 179 (2008) ,

37. Shailesh Kumar, et al.(2012) , " Logical Itemset Mining " IEEE 12th International Conference on Data Mining Workshops, December 2012.

38. Tianyi Wu et al.(2010), " Re-examination of interestingness measures in pattern mining : a unified framework" Data Mining and Knowledge Discovery, Springer, November 2010, Volume 21, Issue 3, pp 371-397

39. Tias Guns et al. (2013), " k- Pattern Set Mining under Constraints" , IEEE Transactions on Knowledge and Data Engineering, Vol. 25, No. 2, February 2013.

40. Vincent S. Tseng et al.(2013), " Efficient Algorithms for mining high utility Itemsets from Transactional Databases" , IEEE Transactions on Knowledge and Data Engineering, Vol. 25, No. 8, August 2013.

41. Vivek Tiwari et al., (2010) , Association rule mining: A Graph Based Approach for mining Frequent Itemsets, 2010 International Conference on Networking and Information Technology, 978-1-4244-7578-0 © 2010 IEEE

42. W.J. Jiang et al.(2005), " A Novel Data Mining Algorithm based on Ant Colony System" , Proceedings of the fourth International Conference on Machine Learning and Cybernatics, Gaungzhou, 18-21 August 2005.

43. William Cheung et al. (2003), " Incremental Mining of Frequent Patterns without candidate Generation or Support Constraint", IDEAS'03, doi.ieeecomputersociety.org/10.1109/IDEAS.2003.1214917.

44. Y. G. Suchayo et al. (2003), " CT-ITL: Efficient Frequent Itemset Mining using a Compressed Prefix Tree with Pattern Growth" , 14th Australasian Database Conference Adelaide, Australia (ADC 2003).

45. Y.J. Tsay, J.Y. Chiang (2004), "An efficient cluster and decomposition algorithm for mining association rules", Information Sciences 160 (1-4) (2004) 161–171.

