Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 2, Issue. 2, February 2013, pg.1 – 5

RESEARCH ARTICLE

An FP-Growth Approach to Mining Association Rules

Rakesh Kumar Soni[1], *Prof. Neetesh Gupta*[2], *Prof. Amit Sinhal*[3]

[1]Department of Information Technology, Technocrats Institute of Technology, RGPV, Bhopal, M.P., India

er06rakeshsoni@gmail.com

[2]Department of Information Technology, Technocrats Institute of Technology, RGPV, Bhopal, M.P., India

gupta_neetesh81@yahoo.com

[3]Department of Information Technology, Technocrats Institute of Technology, RGPV, Bhopal, M.P., India

amit_sinhal@rediffmail.com

Abstract— In the field of data mining researchers implements lots of algorithms for improving the performance of mining. This work is also related to that strategy. This work, introduce an idea in this field. Here use Sampling Technique to convert text document in to the appropriate format. This format contains data in the form of word and topic of word. This format take as a input in FP-Growth algorithm for given support value and get association rules of that transaction data, and after getting association rules apply clustering process and then get clusters for that association rules.

Indexed Terms: - Clustering, Documents, FP-growth, Frequent Document Clustering.

I. INTRODUCTION

Many societies around the world store data in digital format. The digital data rapidly grows day by day; there is need a very efficient way to mine information from them. Frequent item set mining is a core data mining operation and has been broadly deliberate over last decade. It plays an essential role in many significant data mining tasks. Algorithms for frequent item sets mining form the heart of algorithms for a number of other mining problems, including association mining, correlations mining, and mining sequential and emerging patterns. Algorithms for frequent item sets mining have typically been developed for datasets kept in persistent storage. It is very important to develop a reliable technique to cluster huge amount of text data. This dissertation presents an implementation to mine text documents and check similarity between association rules with the help of clustering.

Here data structures that are used in many applications such as Bio informatics, chemical structure, and natural language processing. Nowadays their role rapidly increases in data mining and database management. The graph mining includes searching and indexing graph database, schedule pattern mining techniques and various other applications etc. this techniques mostly used on mining complicated pattern from graph database. The graph mining technique very expansive in sub graph isomorphism has more space and high time complexity compared to other process of data structure [1].

To find the frequent sub graph in a collection of graph is main problem of frequent sub graph mining. If support (occurring frequency) in a given graph is grater then a minimum support then sub graph mining is frequent. Now here, there are two Major mining approaches in frequent sub graph mining: the frequent sub graph mining and a priori based approach [4]. The main difference between two methodologies is how they generate candidate sub graphs.

This work utilized the frequent pattern growth approach (FP-Growth) [5] that find association rules (frequent pattern) and modify it to determine frequent sub graphs. The FP-Growth approach is design to mine frequent item sets mining in market basket analysis [1], which analyse the consumer choice during purchase. It was designed with item set mining not for graph mining and it does not mine frequent subgraph well. Therefore it is necessary to make same variations in that algorithm so it can easily or efficiently use for graph mining.

It is fairly new idea to cluster documents by using an algorithm for association rule mining. Many document mining methods is based on the frequency of keywords for document clustering. In these method document like a vector and there element are keywords with frequency. In many cases, there is not an appropriate way to represent the document ideas. For millions of documents there have to store very large amount of data about keywords. And there is no way to keep relationship in every keyword in every document. Most of the words have several meaning in that document. If they are kept in individual unit, it is very difficult to recognize specific sense of keywords.

This Paper consists of five main sections. The second section describes the Basic Theory of this work. The frame work of this approach is describes in third section, Experimental Result analysis is forth section and finally section five is devoted to conclusion and future work.

II. BASIC THEORY

The formal statement of association rule mining i.e. Let $I=I_1, I_2, \dots, I_m$ be a set of m different attributes, T be transaction that holds a set of items such that $T \subseteq I$, D be a database with different transaction records Ts. An association rule is an consequence in the form of X→Y, where $X, Y \subset I$ are sets of items called item sets, and $X \cap Y = Ø$. X is called originator while Y is called resultant, the rule means X implies Y.

There are two essential basic measures for association rules, support (s) and confidence (c). Since the database is huge and users worry about only those frequently bought items, usually thresholds of support and confidence are predefined by users to drop those rules that are not so sensational or useful. The two thresholds are called minimal support and minimal confidence respectively, further constraints of interesting rules also can be identified by the users. The two basic limitations of Association Rule Mining are: support and confidence.

Support (s) of an association rule is defined as the percentage/fraction of Data that hold X U Y to the total number of records in the data warehouse. The count for each item is increased by one each time the item is come across in different transaction T in database D during the scanning process. For example in a transaction a consumer buys three bottles of beers but retailer only rise the support count number of {beer} by one, in another word if a transaction contains an item then the support count of this item is increased by single. Support(s) is calculated by the following principle:

$$\text{Support}(X\,Y) = \frac{\text{Support count of } X\,Y}{\text{Total number of transaction in D}} \qquad (1.1)$$

From the description retailer can see, support of an item is a statistical significance of an association rule. Suppose the support of an item is 0.1%, it means only 0.1 present of the transaction contain purchasing of this item. The retailer will not pay much attention to such kind of items that are not bought so frequently, obviously a high support is preferred for more exciting association rules. Before the mining process, users can identify the minimum support as a threshold, which means they are only interested in definite association rules that are produced from those item sets whose supports exceed that threshold. However, sometimes even the items sets are not as frequent as defined by the threshold, the association rules produced from them are still important. For example in the superstore some items are very costly, consequently they are not purchased so often as the threshold required, but association rules between those costly items are as important as other frequently bought items to the retailer.

Confidence of an association rule is defined as the percentage/fraction of the number of transactions that cover X U Y to the total number of records that have X, where if the percentages go above the threshold of confidence an interesting association rule X→Y can be formed.

$$\text{Confidence}(X/Y) = \frac{\text{Support}(X\,Y)}{\text{Support}(X)} \qquad (1.2)$$

Confidence is a measure of strength of the association rules, suppose the confidence of the association rule X→Y is 80%, it means that 80% of the transactions that contain X also contain Y together, similarly to ensure the interestingness of the rules identified minimum confidence is also pre-defined by users.

Association rule mining is to finding out association rules that fulfil the pre- defined minimum support and confidence from a given data records [2]. The problem is usually spoiled into two sub problems. One is to find those item sets whose incidences exceed a predefined threshold in the database; those item sets are called frequent or large item sets. The second problem is to create association rules from those large item sets with the constraints of minimal confidence. Suppose one of the enormous item sets is L_k, $L_k= \{I_1, I_2, \cdots, I_{k-1}, I_k\}$, association rules with this item sets are generated in the following technique:

The first rule is $\{I_1, I_2, \cdots, I_{k-1}\} \rightarrow \{I_k\}$, by checking the confidence this rule can be resolute as interesting or not. Then other rule are produced by deleting the last items in the antecedent and inserting it to the consequential, further the confidences of the new rules are checked to determine the interest of them. Those processes iterated until the originator becomes vacant. Since the second sub problem is fairly straight forward, most of the researches focus on the first sub problem.

III. Frame Work of This Approach

Graphs have been used in this dissertation to characterize the documents. Generate a VDG of an individual document using the relationship between its keywords stemming process. That objective is to discovery frequent sub graphs within these VDGs. It believes that these frequent sub graphs will better reproduce the sense of the documents, provided that the documents are focused on a specific subject which is returned by their keywords. The current FP-growth method was calculated to find the frequent items, and it requires general tree traversing to determine those patterns (i.e. item sets). The original FP-growth algorithm, when applied to this problem (which prepared possible by representing edges to items, and VDGs to transactions), creates a set of frequent edges which do not necessarily constitute to a linked sub graph. The VDGs that have cover a list of connected edges. Generation of all possible frequent patterns not only outputs all possible frequent sub graphs but also produces a lot of overhead in the form of all possible sets of frequent, but disconnected, edges. This reasons unnecessary costs during the mining of the VDGs as are only watching for frequent sub graphs and these sets of frequent incoherent edges bear no useful information for document clustering. The time and space required to create and store these disconnected frequent edges have undesirable impact on the overall performance of the FP-growth approach.

These indicate the second problem, which is correlated to the computational cost and the memory requirement of the FP-growth, approach. The FP-growth compresses the unique dataset into a single FP-tree structure. Constructing this tree is not computationally costly, but if the database involves of thousands of VDGs then the tree can become large. Main aim was the second step of the FP-growth. Usually the VDGs hold hundreds of edges. Generating all possible frequent patterns for these edges often generates a large number of frequent patterns, and the memory runs out. FP-tree needs extensive mining to determine the frequent patterns. This mining became widespread based on the minimum support. If the minimum support of a sub graph is very low it can appear infrequently in the VDGs. Also, if the number of documents is high, the cost of creating these sub graphs becomes massive. Moreover, if during the FP-tree mining procedure finds a single branch in the FP-tree, create all possible combination of that branch. This often caused program to run out of memory even with high lowest support. The density factor of the dataset plays a vital rule on the runtime performance of FP-growth. The performance of this algorithm reduces drastically if the resulting FP-tree is very abundant. In this case, it has to create a large number of sub groups and then combine the results returned by each of them. Originally, FP-growth was tried on datasets which have small numbers of items. On the contrary, in case every document has a large amount of edges which found the VDG. To reach high quality clusters in frequent pattern-based clustering, the support is usually preserved very low (i.e., 3-5%). With the normal FP-growth approach, often ran out of memory (A machine with GB of RAM) even when the minimum support was as high as 70% to 80%.

IV. Experimental Result Analysis

Experimental Results of Modified FP Growth Approach

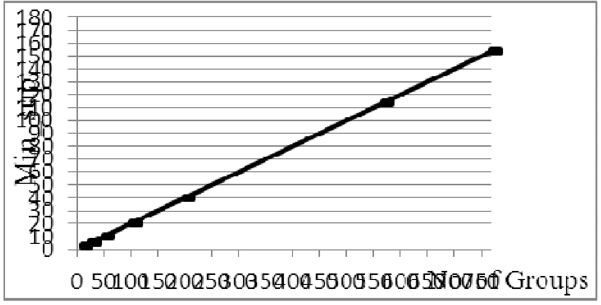

Figure 1 Graph for minimum support of FP-Growth Algorithm.

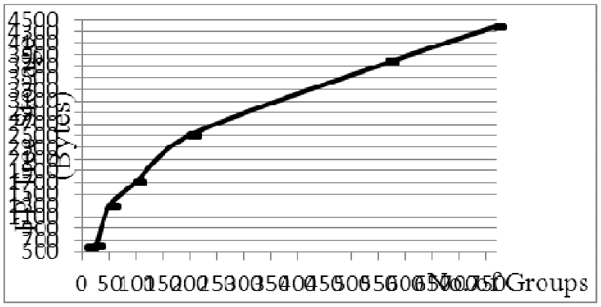

Figure 2 Graph for FP-Tree Storage in Bytes

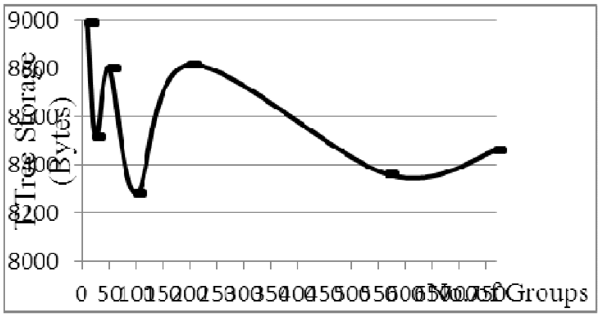

Figure 3 Graph for T-Tree Storage in Bytes

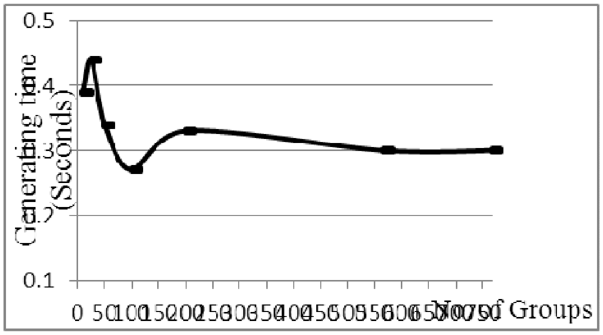

Figure 4 Graph for Generating time in second of FP-Growth

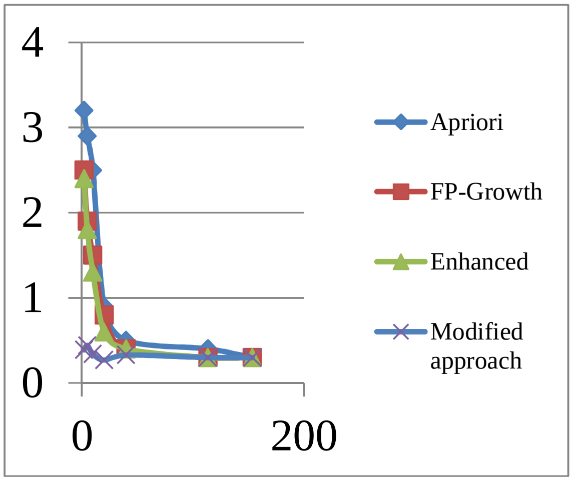

Figure 5 A comparisons between result of Execution Time and Support from another implementation

.

V. CONCLUSION

In Data mining document clustering is very active research area.to find suitable information system much suitable idea has been implemented in document clustering. It is very challenging task to find human-like clustering. In this work, a graph based clustering with affinity propagation. That find a new way to clustering document based more on the keywords they contain document based clustering techniques mostly depend on the keywords. The work is modifying the FP-mining algorithm to find the frequent sub graph with clustering affinity propagation in graph.

It can evaluate some techniques for computing the combination of large scale paths which can improve the performance of mining algorithms. In future will plan to study more application related to that feature and try to implement batter way and improve their performance.

REFERENCES

[1] Jiawei Han and Micheline Kamber, “Data Mining: Concepts and Techniques”, 2nd edition, The Morgan Kaufmann Series in Data Management Systems, Jim Gray, Series Editor Morgan Kaufmann Publishers, March 2006.

[2] M. H. Margahny, A. A. Mitwaly, “Fast Algorithm for Mining Association Rules”, AIML Artificial Intelligence and Machine Learning Conference on CICC, Cairo, Egypt, December 2005.

[3] RakeshAgrawal, RamakrishnanSrikant, “Fast algorithms for mining association rules”, VLDB Proc. 20th Int. Conf. Very Large Data Bases, volume 1215, pp. 487-499, April 2005.

[4] Moti Cohen, Ehud Gudes, “Diagonally Sub graphs Pattern Mining”, Paris, France. ACM, DMKD Data Mining and Knowledge Discovery, June 2004.

[5] Jiawei Han, Hong Cheng, Dong Xin, Xifeng Yan, “Frequent pattern mining: current status and future directions”, USA, Springer Science, Business Media, January 2007.

[6] Bharat Gupta, Dr. Deepak Garg, “FP-Tree Based Algorithms Analysis: FPGrowth, COFI-Tree and CT-PRO”, IJCSE International Journal on Computer Science and Engineering, Volume 3, pp-2691-2699, July 2011.

[7] Kuldeep Malik, NeerajRaheja, PuneetGarg “Enhanced FP-Growth Algorithm”, IJCEM International Journal of Computational Engineering & Management, Volume 12, pp-54-56, April 2011.

[8] M Suman, T Anuradha, K Gowtham, A Ramakrishna, “A Frequent Pattern Mining Algorithm Based on Fp-Tree Structure and apriori Algorithm”, IJERA International Journal of Engineering Research and Applications, Volume 2, issue 1, pp. 114-116, February 2012.

[9] AimanMoyaid Said, Dr. P D D. Dominic, Dr. Azween B Abdullah, “A Comparative Study of FP-growth Variations”, IJCSNS International Journal of Computer Science and Network Security, Volume 9, issue 5, pp. 266-272, May 2009.

[10] E.R.Naganathan, S.Narayanan, K.Rameshkumar, “Fp-Growth Based New Normalization technique For Subgraph Ranking”, IJDMS International Journal of Database Management Systems, Volume 3, issue 1, pp. 81-91, February 2011.

[11] Varun Krishna, N. N. R. RangaSuri, G. Athithan, “A comparative survey of algorithms for frequent subgraph discovery”, Artificial Intelligence and Robotics, Current Science, volume 100, January 2011.

[12] J. Chandrika, K. R. Ananda Kumar, “State of The Art Algorithms For Frequent Item Mining In Data Streams”, IJCSC International Journal of Computer Science and Communication, Volume 2, No. 2, pp. 479-484, December 2011.

[13] Charu C. Aggarwal, ChengxiangZhai, “Mining Text Data”, first edition, Kluwer Academic Publishers Boston/Dordrecht/London, USA, November 2011.

[14] GostaGrahne, Jianfei Zhu, “Fast Algorithms for Frequent Itemset Mining Using FP-Trees”, IEEE Transactions on Knowledge and Data Engineering, volume 17, pp. 1347-1362, October 2005.

[15] M. ShahriarHossain and Dr. Rafal A. Angryk, GDClust: “A Graph-Based Document Clustering Technique”, ICDM International Conference on Data Mining, USA, pp. 417-422, 2007.