Volume 6, Issue 6, June 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Mining High Utility Itemsets using Hadoop

Parul Dubey
ME Scholar,
G.S. Moze College of Engineering,
Pune, Maharashtra, India

Ratnaraja Kumar
Professor, HOD Computer Science,
G.S. Moze College of Engineering,
Pune, Maharashtra, India

Abstract: ***In thefield of data mining, utility mining emerges as an important topic therefore mining the high utility itemsets from databases refers to finding the itemsets with high profits. Mining high utility itemsets from a transactional database is referred to the discovery of itemsets with high utility. In the past few years a large number of algorithms have been proposed for mining high utility itemsets from a transactional database. The problem with all those algorithm is the production of a large number of high utility itemsets. Hence the mining performance degrades and it is in terms of execution time and space requirement. Therefore algorithms named UP Growth and UP Growth+ are implemented which requires just two scanning and generate less number of high utility itemset. Performance is further enhanced by implementing the algorithms in Hadoop, so that system with less memory or data set which requires more memory can be analyzed in low memory based system with the help of distributed file system. Overall the performance is upgraded with respect to execution time as well as space. The performance of UP-Growth and UP-Growth+ is compared with the progressive methods on many varieties of each real and artificial information sets. In experimental studies we are going to compare the performances of existing algorithms UP-Growth and UP-Growth+ against the improve UP-Growth and UP-Growth+ victimization dimension reduction with Hadoop.***

Keywords: Dataset mining, Hadoop, Dimensional reduction, Item sets, Map reduce framework, Transactional dataset, UP-Growth, UP-Growth+, Up-Tree.

I. INTRODUCTION

Data mining is the process of revealing nontrivial, previously unknown and potentially useful information from large databases. Discovering helpful patterns hidden in a huge database plays an important role in many data processing tasks, few examples are frequent pattern mining, weighted frequent pattern mining, and high utility pattern mining. Among these, frequent pattern mining could be an elementary analysis topic these has been applied to completely different forms of databases, like transactional databases, streaming databases and statistic databases.

Two algorithms namely Utility Pattern growth (UP-Growth) and UP-Growth+, and a compact tree structure referred to as utility pattern tree (UP-Tree), are used for finding high-utility itemsets and maintaining important data associated with utility patterns.Identifying High Utility Itemsets from a Transactional Database is the major focus area to improve the performanceof these algorithms. High-utility itemsets can be generated from UP-Tree; this can be done with just two scans of original databases.

Secondly, these algorithms performance will be enhanced further by implementing it on haddop. Their performances are also compared. Finding high utility itemset in a transactional database is a difficult task if there is a low memory system.

II. LITERATURE SURVEY

In this section we have presented the survey of different methods for mining high utility item sets from the transactional datasets.Firstly,R. Agrawal and R. Srikant, Fast Algorithms for Mining Association Rules, [3] had discussed a well-known algorithms for mining association rules is Apriori, which is the pioneer for efficiently mining association rules from large databases.The Apriori and AprioriTid algorithms proposed by authors differ fundamentally from the AIS and SETM, these algorithms in which candidate itemsets are counted in a pass and then candidates are generated.

Secondly, J. Han, J. Pei, and Y. Yin, Mining Frequent Patterns without Candidate Generation, [4] had discussed Pattern growth-based association rule mining algorithms, such as FP-Growth which was proposed afterward.Thirdly,Cai et al. and Tao et al. first proposed the concept of weighted item and weighted association rules [5].The framework of weighted association rules doesn't have downward closure property, mining performance cannot be improved in that case For finding a solution to this problem, Tao et al. proposed the concept of weighted downward closure property[12].

Nextly, Liu et al. proposed an algorithm named Two- Phase [8] which is mainly composed of two mining phases. In phase I, this employs an Apriority based level-wise method for enumerating the HTWUIs. Liet al. [7] proposed an Isolated Items Discarding Strategy (IIDS) to reduce the number of candidates. By pruning isolated items during level-wise search, the number of candidate item sets for HTWUIs can be further reduced.

Ahmed et al. [13] proposed tree based algorithm, named IHUP. In this, a tree based structure called IHUP-Tree is used to maintain the information about item sets and their utilities.Each node of an IHUP-Tree consists of three thingsan items name, a TWU value and a support count. IHUPalgorithm has the following three steps: 1) construction of IHUP-Tree, 2) generation of HTWUIs, and 3) identification of high utility item sets. In step 1, items in transactions are re-arranged in a fixed order such as lexicographic order, support descending order or TWU descending order. Then these rearranged transactions are inserted into an IHUP-Tree.

III. PROPOSED SYSTEM

Many of proposed ways for high utility mining from large datasets are studied above in the previous section,the literature survey. However, performance is diminished and there are a large set of PHUIs in all the studies, it consumes upper time interval. The ways in [1] exceed the progressive algorithms virtually altogether cases in every real and artificial information set.Figure 1 showsthe proposed system block diagram and details of algorithm proposed.

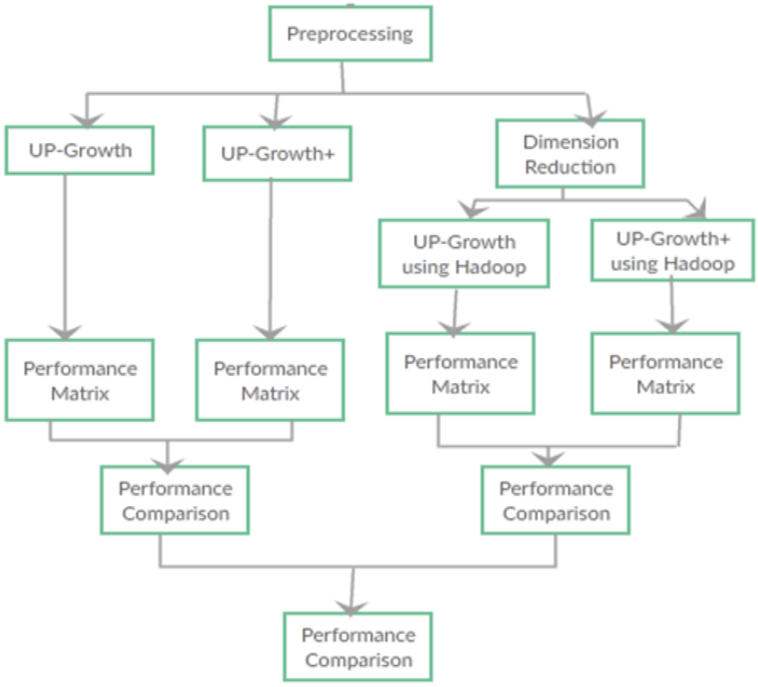

Fig 1. System Architecture

Hadoop and Map Reduce

Hadoop is an open-source framework that allows to store and process big data in a distributed environment,this is done across clusters of computers with the help of simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage.

MapReduce is considered as a processing technique and a program model for distributed computing. It contains two important tasks,i.e. Map and Reduce. Map takes a set of data and converts it into another set of data. In this individual elements are broken down into tuples (key/value pairs). Nextly, reduce task, which takes the output from a map as an input at the same time it combines those data tuples into a smaller set of tuples. The reduce task is always performed after the map job.

In our proposed system we are going to develop a data structure named UP-tree and then going to implement UP-growth and UP-growth+ algorithm using map and reduce procedure.

Constructing a Global UP-Tree

This can be done with two scans of the original database. In the first scan, Firstly, TU is calculated at the same time, TWU of each single item is also accumulated. If its TWU is less than the minimum utility thresholdthen it is considered unpromising. An item ip is called a promising item if TWU (ip) |min_util|. New TU after pruning unpromising items is called reorganized transaction utility (abbreviated as RTU).After this begins the second scanning.Strategy DGU uses RTU to overestimate the utilities of item sets. As the utilities of unpromising items are excluded, RTU must be no larger than TWU. Hence, the number of PHUIs with DGU must be no more than that of HTWUIs generated with TWU. DGU is effective especially when transactions contain lots of unpromising items. The search space can be divided into smaller subspaces as we need to apply divide and conquer. By applying strategy DGN, theutilities are further reduced for the nodes that are closer to the root .The more items a transaction contains, the moreutilities can be discarded by DGN.

UP-Growth

The number of PHUIs can be further reduced by decreasing the overestimated utilities, using the following two strategies in UP-Growth algorithm.

Discarding Local Unpromising Items during Constructing a Local UP-Tree

The common method for generating patterns in tree based algorithms contains following three steps: (1) Generate conditional pattern bases by tracing the paths in the original tree, (2) construct conditional trees (also called local trees in this paper) by the information in conditional pattern bases and (3) mine patterns from the conditional trees [1]. But strategies DGU and DGN cannot be applied into conditional UP-Trees. A Reason behind this is actualutilities of items in different transactions are not maintained in a global UP-Tree. We maintain a minimum item utility table to keep minimum item utilities for all global promising items.

Decreasing Local Node Utilities during Constructing a Local UP-Tree

Since $\{i_m\}$-Tree must not contain the information about the items below i_m in the original UP-Tree, we can discard the utilities of descendant nodes related to i_m in the original UP-Tree while building {im}-Tree.[21] (Here, original UP-Treemeans the UP-Tree which is used to generate $\{i_m\}$-Tree.) Path utility of item i_k in $\{i_m\}$-CPB is denotedas pu(ik,$\{i_m\}$-CPB).The same as DLU, DLN can be recognized as local version of DGN [21].

UP-Growth: Mining a UP-Tree by Applying DLU and DLN

Firstly we trace the node link to root of the UP-Tree to get paths related to i_m in UP-Tree correspondingto the item i_m, .All retrieved paths, their path utilities and support counts are collected into i_m's conditionalpattern base(CPB)[21]. A conditional UP-Tree can be constructed by two scans of aCPB. For the first scan, local promising and unpromising items are learned by summing the path utility for each item in the conditionalpattern base [21]. Then, DLU is applied, in order to reduce overestimated utilities during the second scan of theCPB. When a path is retrieved, unpromising items and their estimated utilities are eliminated from the path and its path utility [21]. Then the path is reorganized, this is done by the descending order of path utility of the items in the CPB.

UP-Growth+

The overestimated utilities can be closer to their actual utilities by eliminating the estimated utilities that are closer to actual utilities of unpromising items and descendant nodes, hence UP-Growth+ is proposed. Minimal node utilities in each path are used to make the estimated pruning values closer to real utility values. Firstly, we need to add anelement,N.mnu, into each node of UP-Tree. N.mnu is minimal node utility ofN. When N is traced, N.mnukeeps track of the minimal value of N.name's utility in different transactions.[21] and If N.mnu is larger than u (N.name, Tcurrent), N.mnu is set to u(N.name, Tcurrent).

IV. CONCLUSION

In this paper, we have explained two algorithm namely UP-Growth and UP-Growth+ and given the idea of implementing the same with Hadoop for mining high utility itemsets from transaction databases. In this way it can be implemented in low memory based system as well.

REFERENCES

[1] Vincent S. Tseng, Bai-EnShie, Cheng-Wei Wu, and Philip S. Yu, Fellow, IEEE, EfficientAlgorithms for Mining High Utility Item sets from Transactional Databases, IEEE TRANSACTION ON KNOWLEDGE AND DATA ENGINEERING, VOL. 25, NO. 8, AUGUST 2013.

[2] R. Agrawal and R. Srikant, Fast Algorithms for Mining Association Rules, Proc. 20th Intl Conf. Very Large Data Bases (VLDB), pp. 487-499, 1994

[3] C.F. Ahmed, S.K. Tanbeer, B.-S. Jeong, andY.-K. Lee, Efficient Tree Structures for High UtilityPattern Mining in Incremental Databases, IEEE Trans.Knowledge and Data Eng., vol. 21, no. 12, pp. 1708-1721, Dec. 2009.

[4] C.H. Cai, A.W.C. Fu, C.H. Cheng, and W.W.Kwong, Mining Association Rules with WeightedItems, Proc. Intl Database Eng. and Applications Symp.(IDEAS 98), pp. 68-77, 1998.

[5] Han, J. Pei, and Y. Yin, Mining FrequentPatterns without Candidate Generation, Proc. ACMSIGMODIntl Conf. Management of Data, pp. 1-12,2000

[6] S.C. Lee, J. Paik, J. Ok, I. Song, and U.M. Kim,Efficient Mining of User Behaviors by Temporal MobileAccess Patterns, Intl J. Computer Science Security, vol.7, no. 2, pp. 285-291, 2007..

[7] H.F. Li, H.Y. Huang, Y.C. Chen, Y.J. Liu, andS.Y. Lee, Fast and Memory Efficient Mining of HighUtility Itemsets in Data Streams, Proc. IEEE EighthIntl Conf. on Data Mining, pp. 881- 886, 2008.

[8] Y.-C. Li, J.-S. Yeh, and C.-C. Chang, IsolatedItems Discarding Strategy for Discovering High UtilityItemsets, Data and Knowledge Eng., vol. 64, no. 1, pp.198-217, Jan. 2008..

[9] C.H. Lin, D.Y. Chiu, Y.H. Wu, and A.L.P. Chen,Mining Frequent Itemsets from Data Streams with aTime-Sensitive Sliding Window, Proc. SIAM Intl Conf.Data Mining (SDM 05), 2005.

[10] Y. Liu, W. Liao, and A. Choudhary, A Fast HighUtility Itemsets Mining Algorithm, Proc. Utility-BasedData Mining Workshop, 2005.

[11] F. Tao, F. Murtagh, and M. Farid, WeightedAssociation Rule Mining Using Weighted Support andSignificance Framework, Proc. ACM SIGKDD Conf.Knowledge Discovery and Data Mining (KDD 03), pp.661-666, 2003.

[12] H.F. Li, H.Y. Huang, Y.C. Chen, Y.J. Liu, andS.Y. Lee, Fast and Memory Efficient Mining of HighUtility Item sets in Data Streams, Proc. IEEE EighthIntl Conf. on Data Mining, pp. 881-886, 2008.

[13] Y.-C. Li, J.-S. Yeh, and C.-C. Chang, IsolatedItems Discarding Strategy for Discovering High UtilityItem sets, Data and Knowledge Eng., vol. 64, no. 1,pp. 198-217, Jan. 2008.

[14] C.H. Lin, D.Y. Chiu, Y.H. Wu, and A.L.P.Chen, Mining Frequent Item sets from Data Streamswith a Time-Sensitive Sliding Window, Proc. SIAMIntl Conf. Data Mining (SDM 05), 2005.

[15] Y. Liu, W. Liao, and A. Choudhary, A FastHigh Utility Item sets Mining Algorithm, Proc. Utility Based Data Mining Workshop, 2005.

[16] E. Omiecinski, Alternative interesting measuresfor mining associations, The IEEE Transaction onKnowledge and Data Engineering, Vol. 15, no. 1, pp.57-69, 2003.

[17] J. Dean, and S. Ghemawa, MapReduce:Simplified Data Processing on Large Clusters, OSDI,2004.

[18] H. Dutta, and J. Demme, Distributed Storageof Large Scale Multidimensional EEG Data usingHadoop/HBase, Grid and Cloud Database Management,New York City: Springer; 2011.

[19] M.R. Karim, B.S. Jeong, and H.J. Choi,A MapReduce Framework forMining MaximalContiguous Frequent Patterns in Large DNA SequenceDatasets, IETE Technical Review, Vol. 29, no. 2, pp.162-8, Mar-Apr, 2012.

[20] M.R. Karim, and B.S. Jeong, A Paralleland Distributed Programming Model for MiningCorrelated, AssociatedCorrelated Independent Patternsin Transactional Databases Using MapReduce onHadoop, Proc. of the 6th Intl. Conf. on CUTE, pp.2716, 2011.

[21] D.Sathyavani ,Efficient Algorithm for Finding High UtilityItemsets from Large TransactionalDatabases Using R-Hashing Technique, IJIRCCE, Vol. 2, Special Issue 3, July 2014,PP.137-143

