International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 22 (2017) pp. 12169-12175
© Research India Publications. http://www.ripublication.com

Study of Association Rule Mining for Discovery of Frequent Item Sets on Big Data Sets

Chandaka Babi[1], Dr. Mandapati Venkateswara Rao[2] and Dr. Vedula Venkateswara Rao[3]

[1]Research Scholar, [2]Professor, [3]Professor
[1,2]Department of Information Technology, Gitam Institute of Technology, Visakhapatnam, Andhra Pradesh, India.
[3]Department of Computer Science and Engineering, Sri Vasavi Engineering College, Tadepalligudem, Andhra Pradesh, India.

[1,2,3]Orcid Id: 0000-0001-1166-944x, 0000-0002-7598-3473, 0000-0003-0131-4944

Abstract

Frequent pattern mining is an significant research area in data mining. Since its introduction, it has drawn consideration of many researchers. Data generation and collection diagonally all areas enhance in size exponentially. Knowledge discovery and decision making requires the capability to process and extract imminent from “Big” Data in a scalable and efficient manner. Data mining is the application of sophisticated analysis to large amounts of data in order to discover new knowledge in the form of patterns, trends, and associations. With the beginning of the World Wide Web, the quantity of data stored and available by electronic means has developed extremely and the process of knowledge discovery (data mining) from this data has become very central for the business and scientific research communities. Many algorithms have been projected to mine frequent Item sets. Well-liked algorithms include level-wise Apriori based algorithms, tree based algorithms, and hyperlinked array structure based algorithms. While these algorithms are accepted and useful due to some good possessions, they also experience from some problems such as multiple database scans, recursive tree constructions, or multiple hyperlink adjustments. In the current era of big data, *high volumes* of a wide *variety* of *valuable* data of different *veracities* can be easily collected or generated at high *velocity* in a range of real-life applications. The problem of mining frequent queries in a relational database defined over a star schema is not easy even when we deal with only one table, because, on the one hand, the size of the search space is huge (because encompassing all possible queries that can be addressed to a given database), and on the other hand, testing whether two queries are equivalent (which entails redundant support computations) is NP-Complete. Therefore, the problem is even more difficult when they are applied to Big Data. In this paper we focus on handling high volumes of big data to generate frequent Item sets. In this paper, we examine in details the problems related to the Frequent Pattern Mining (FPM) in distributed and large data sets and present a general framework for adapting an exact Frequent Pattern Mining algorithm

Keywords: Data Mining, Frequent Item Sets, Association Rules, Big Data Sets, Frequent Pattern Mining, NP Complete.

INTRODUCTION

Over the past two decades, with progress in storage media technology, storage devices have turn into larger and more efficiently viable. As a consequence, businesses, large corporations, etc., have started storing and archiving various types of data in the form of large databases. The principle of storing data is two-fold: first, to access these data in the future, and second, for analysis and discovery of co-relation or relationships among data items stored in database. The job of finding association or co-relation among the data items is known as association rule mining. The main stimulus comes from market basket analysis [Fan Jiang et al 2014]. When in a supermarket a customer approaches to buy some items, how likely is she to buy some other specific items or how often customers are purchasing a set of items together?. In recent times, researchers have started discovering rules for data where existence of an item is not sure [Kun He et al. 2015]. In this case, an item is present in transaction by some possibility determine.

Data mining has involved a great deal of interest in the information industry and in society as intact in current years, due to the wide availability of massive amounts of data and the impending need for spiraling such data into practical information and knowledge. The information and knowledge gained can be used for applications varying from market analysis, fraud detection, and customer retention, to production control and science exploration. Frequent pattern mining is an vital area of Data mining research. The frequent patterns are patterns (such as item sets, subsequences, or substructures) that emerge in a data set frequently. For instance, a set of items, such as milk and bread that appear frequently together in a transaction data set is a *frequent item set*. A subsequence, such as buying first a PC, then a digital camera, and then a memory card, if it occurs frequently in a shopping history database, is a *frequent sequential pattern*. A *substructure* can refer to different structural forms, such as sub graphs, sub trees, or sub lattices, which may be combined with item sets or subsequences. If a substructure occurs frequently, it is called a *frequent structured pattern*. Finding such frequent patterns plays an essential role in mining associations, correlations, and many other interesting relationships among data. Moreover, it helps in data classification, clustering, and other data mining tasks as well.

In the Big Data era, we all, in one form or another, contribute in generating data. Big Data is heterogeneous. It can be **structured**, which is generated by applications like Customer Relationship Management (CRM) and Enterprise Resource Planning (ERP) systems and naturally stored in rows and columns with definite schemas. It can be **semi-structured**, which is produced by sensors, web feeds, event monitors, stock market feeds, and network and security systems. Semi-structured data typically have meta-data that depicts their structure; nevertheless this organization does not always fit in rows and columns. Big Data can also be **unstructured**, which is naturally produced by people in forms such as social media, text documents, videos, audio and images. Beside this variety of data formats, Big Data is produced in **enormous volumes** at a **rapid velocity** with no clear way of specifying the **veracity** of it. With such attributes, data has outgrown the capability to be stored and practiced by many conventional systems [Cheikh et al. 2013]. The price of data is realized from first to last insights, taking into deliberation that the usefulness of some data points takes a rain check very quickly. Gradually more, businesses success has turn into dependent on how rapidly and efficiently they can turn the peta-bytes of data they collect into actionable information [Bart Goethals et al. 2012].

MOTIVATION

Big Data Analytics present businesses the way to find out hidden patterns in such data and use these patterns to guess the possibility of future events. Analytics can be Descriptive which is used to recapitulate what happened. It can be Predictive which employs a variety of statistical, modeling, data mining, and machine learning methods to study recent and historical data, by this means allowing businesses to build predictions about the future. There is also an promising form of Analytics called Prescriptive Analytics that suggests one or more courses of action and demonstrates the likely result of each decision. Businesses need to defeat a number of challenges to harvest the benefits of Analytics. The goal n this paper focus at concentrate on these challenges to empower businesses to do Predictive Big Data Analytics to formulate predictions about the future and make use of them to suggest courses of action.

Big Data and Characteristics

The data is collected and stored in every minute, every hour and every day in an organization or institute and is available in large quantity. But the quantity of data is not of significance but what the organizations do with these data to make out information that can be helpful for them. This can be done by analyzing the data to recognize insights or critical information that can help the organization to make useful decisions for their growth. The term big data explains a huge volume of data that is on hand in both structured and in unstructured formats. Even though the idea of big data is a new term, the process of collecting the data, storing them in large amounts and analyzing them to collect new information is a little that has been done since long before big data has been used. The characteristics of big data can be explained using 3 V’s such as (1) Volume, (2) Velocity and (3) Variety. The applications of big data include areas such as health care, telecom, finance, etc.

Data Mining in Big Data

Big Data mining works with a huge amount of data that is stored in the data warehouses and databases. The perception of big data mining can be used to extract or recognize the interesting patterns and information from these great data. Many data mining techniques are on hand that can be functional to the big data. They are classification, clustering, association rules, prediction, estimation, documentation and description. The investigation about these techniques has been outsized since long ago. Many algorithms have been applied in each of the data mining techniques and this also applies to big data. One such well identified technique that is applied is the association rule mining in big data. This is a most efficient data mining technique that is used to determine a range of hidden patterns and information from large databases. Here the relationships between the various characteristics of the data are recognized using the association rule mining algorithm. Some fundamental types of association rule mining algorithms are the Apriori algorithm, Distributed algorithm and Parallel algorithm.

Association Rule Mining

The Association Rule Mining (ARM) [Yifan Chen et al. 2015] in data mining is a accepted approach that is used to analyze the given dataset to find out interesting patterns or relations among the various items in the dataset. The idea of well-built association rules was initially used by Alfredo et al. [2013] to make out the different association rules between the items that are traded during a large scale transaction database gathered from a supermarket by means of a point system. The association among the items is recognized based on the purchase pattern. The ARM technique produces a set of association rules that are widespread among the various items of the given dataset based on the numeral occurrences of these items combination in the dataset.

Literature Survey

The Map Reduce can be used to devise the existing sequential algorithms into parallel algorithms that can be used to tackle large amounts of data with in less time and consequently this is functional for association rule mining [6]. A number of existing methods have been discussed as given below.

Advancement in Association Rule Mining

Yang et al. proposed a Map Reduce based programming model for invention of association rules in Hadoop framework to handle large volumes of data. The Apriori algorithm is used as the essential association rule generation technique. But the customary Apriori algorithm is time intense and it captures consumption of more time particularly when dealing with numerous candidate sets. To trounce this subject, they employed the enhanced Apriori algorithm that is parallelized using the Hadoop framework to save time. The use of Hadoop for association rule generation offered new research focus in forthcoming years. The improved Apriori algorithm is suggested by Yang et al. that largely works by means of the Map Reduce concept to handle large data by making use of the various nodes in Hadoop platform. Lin et al. proposed a comparable method for association rule generation by via the same Apriori approach for frequent item set generation in Hadoop platform using the Map Reduce approach. The mining process is executed in a rapid manner by employing the parallelized mining technique throughout frequent item set generation. But parallelization cannot be handled efficiently. For this purpose the Map Reduce is used. They future a parallelization algorithm in Map Reduce that achieves enhances improvement than the earlier algorithms in terms of speed and efficiency in rule generation. That is, the contrast of results obtained here shows better performance in terms of both speed and the rule generation accuracy with existing algorithms. Riondata et al. proposed a randomized algorithm for association rules mining that is implemented using a parallel approach in Map Reduce framework.

Applications of Big Data

Healthcare: With the advent of the big data the diagnosis of the diseases has become simpler. Diseases can be diagnosed at a very early stage.

Governmental organizations: Many of the governmental organizations use the big data as a tool to overcome many problems faced by the government. The big data is much useful in the election campaigns.

Manufacturing Industries: The use of the big data in the manufacturing industries helps in improving the quality of the product. A huge amount of the sensory data and the historical data makes up the big data which helps in the manufacturing.

Media: Big data plays an important role in the media. It provides more technologies to gather much information from the targeted consumers which mainly help the advertisers for the marketing purposes.

FREQUENT PATTERN MINING: SUMMARY

Preliminaries

Let I ={i1; i2; …….in} be a set of items. An item set C is a subset of I. We denote by |C| its length or size, i.e. the number of items in C. Given a list of transactions T, where each transaction T 2 T is an item set, |T| denotes the total number of transactions. Transactions are generally identified by a transaction id (tid). The support of C is the proportion of transactions in T that contain C. The support count, or frequency of C is the number of transactions in T that contain C. An item set is said to be a frequent item set if it has a support greater than some user defined minimum support threshold **σ.**

The problem of frequent pattern mining (FPM) is formally defined as follows. Its specialization for the frequent item set mining, frequent sequence mining (FSM), and frequent graph mining (FGM) is straight-forward.

Definition: Given a pattern container P and a user-specified parameter **σ** (0 <= **σ** <=1), find all sub-patterns each of which is supported by at least [**σ** |P|] patterns in P.

Basic Data Mining Methodologies

Many complicated frequent item set mining techniques have been urbanized over the years. Two nucleus methodologies materialize from these methods for tumbling computational cost. The first intend to prune the candidate frequent item set search space, while the second center on reducing the numeral of comparisons essential in shaping item set support.

Candidate Generation

A brute-force approach to establish frequent item sets in a set of transactions is to calculate the support for every potential candidate item set. Given the set of items I and a partial order with respect to the subset operator, one can describe all possible candidate item sets by an item set lattice, in which nodes represent item sets and edges correspond to the subset relation. Figure1 shows the item set lattice containing candidate item sets for example transactions denoted in Table 1.

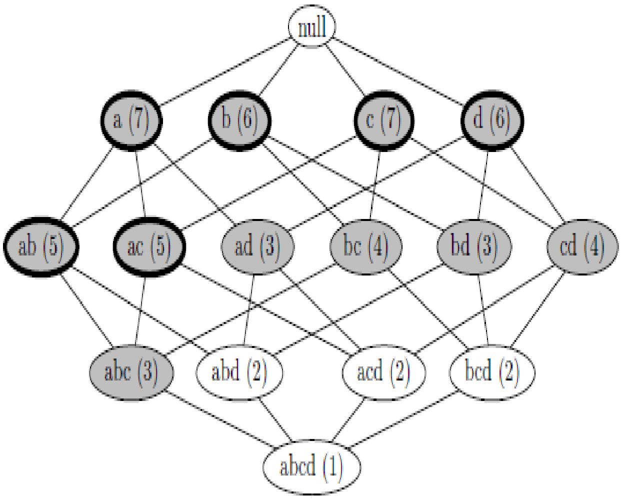

Figure 1: An item set lattice for the set of items I = {a,b,c,d}.

Table 1: Transactions with items from the set I = {a, b, c, d}

Tid	**Items**
1	a, b, c
2	a, b, c
3	a, b ,d
4	a, b
5	a, c
6	a, c, d
7	c. d
8	b, c, d
9	a, b, c, d
10	d

Pattern Growth

Apriori-based algorithms route candidates in a breath first search method, decaying the item set lattice into level-wise item set-size based equivalence classes: k-item sets must be processed before (k + 1)-item sets. presuming a lexicographic ordering of item set items, the search space can also be festering into prefix based and suffix based equivalence classes. Figures 2 and 3 show equivalence classes for 1-length item set prefixes and 1-length item set suffixes, respectively, for our test database. Once frequent 1-itemsets are discovered, their equivalence classes can be mined independently. Patterns are grown by appending (prepending) appropriate items that follow (precede) the parent's last (first) item in lexicographic order.

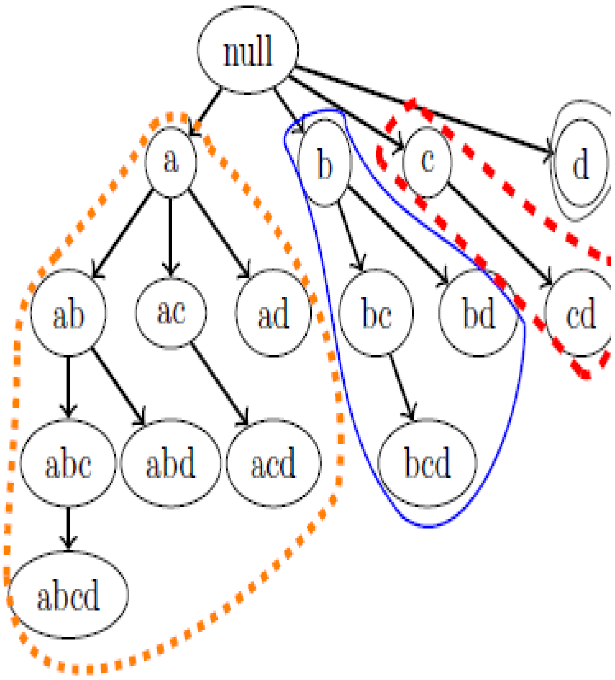

Figure 2: Prefix tree showing prefix-based 1-length equivalence classes in the item set lattice for I = {a, b, c, d}

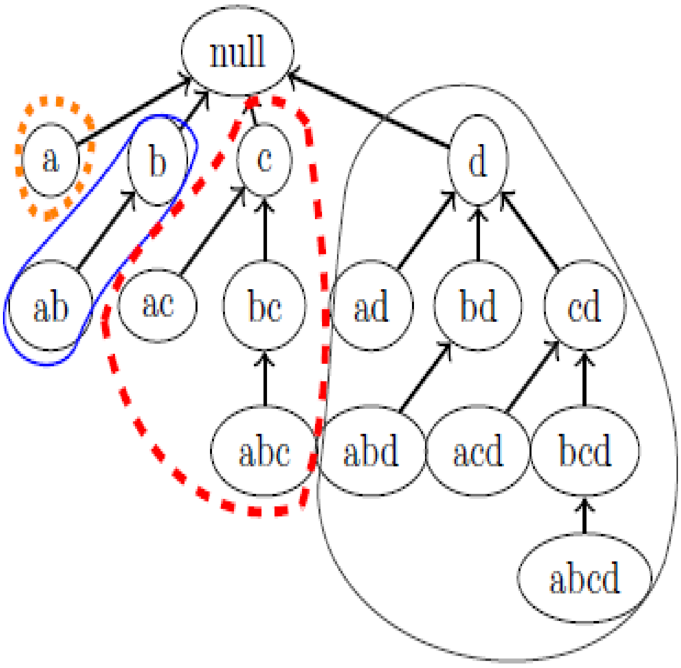

Figure 3: Suffix tree showing suffix-based 1-length equivalence classes in the item set lattice for I = {a, b, c, d}

Models for Big Data Computation

The challenge of operational with Big Data is two-fold. First, dataset sizes have increased much faster than the on hand memory of a workstation. The second confront is the computation time required to find a solution. Computational parallelism is an essential tool for managing the massive scale of today's data. It not only allows one to operate on more data than could not on a single machine, but also gives speedup opportunities for computationally intensive applications.

Philosophy of Parallel Algorithms

Designing a parallel algorithm is not an easy prospect. In addition to all of the challenges associated with serial algorithm design, there are a host of issues specific to parallel computation that must be considered. As one might imagine, extending the serial FIM methods to parallel systems need not be difficult. For example, a serial candidate generation based algorithm can be made parallel by replicating the list of transactions T at each process, and having each process compute the support for a subset of candidate item sets in a globally accessible hash tree. These “direct" extensions however; rely on assumptions like unlimited process memory and concurrent read / concurrent write architecture, which ignore the three challenges.

One of the key factors in choosing to use parallel algorithms in lieu of their serial counterparts is data that is too large to fit in memory on a single workstation.

Memory scalability is essential when working with Big Data as it allows an application to cope with very large datasets by increasing parallelism.

A second important challenge in designing a successful parallel algorithm is to decompose the problem into a set of tasks, where each task represents a unit of work, tasks are independent and can be executed concurrently, in parallel. Given these independent tasks, one must devise a work partitioning, or static load balancing strategy, to assign work to each process. A good work partitioning attempts to assign equal amounts of work to all processes; all processes can finish their computation at the same time.

Hadoop Distributed File System

HDFS is mainly used for storage and computation of large amount of data across various clusters of servers. The HDFS uses master/slave architecture. The metadata is stored on a server called the name node which acts as the master. The data node is a server which stores the application data. There is more than one data node which acts as the slave.

The following diagram shows arcetecture of HDFS

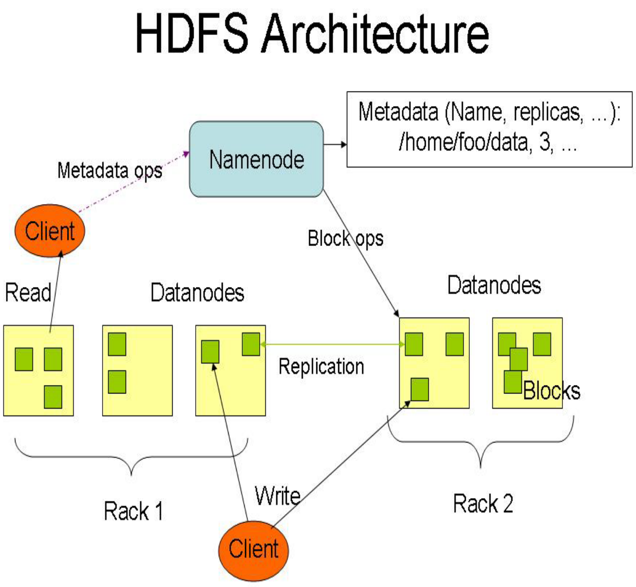

Figure 4: Arcetecture of Hadoop Distributed File System (HDFS)

Map reduce

Hadoop Map reduce framework is used for parallel processing of large amounts of data. It involves two main tasks, the map task and the reduce task.

Map task

It takes the input data and produces the key/value pairs. The key/value pair is the output of the map task. The map task is performed by the mapper.

Reduce task

The reducer obtains the output data from the mapper and process the data. The processed data is stored in the HDFS. Reducer has three main tasks which are the shuffling, sorting and reducing.

The following diagram describes function of Hadoop Computing Model

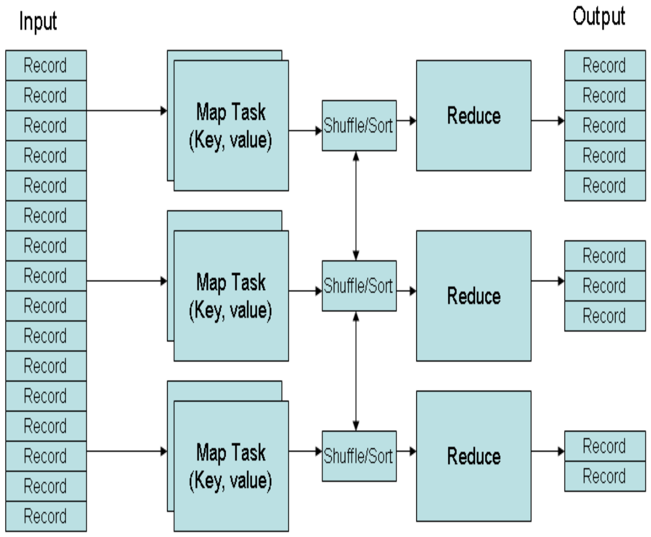

Figure 5: Hadoop Computing Model

The following diagram explains how to join Large and small Data Sets

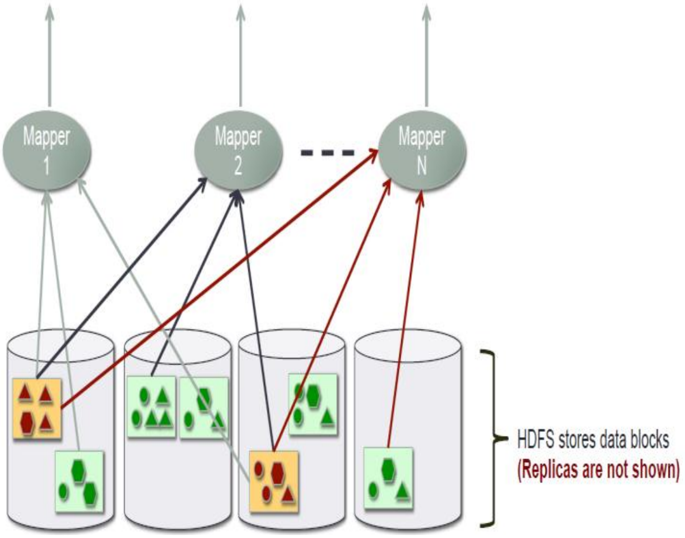

Figure 6: Joining Large and Small Data Sets

CONCLUSION

Many well-organized sequential algorithms have been developed for answering the frequent pattern mining problem.

Until now they do not scale to the type of data we are presented with today, the so-called \Big Data". In this paper, we gave an overview of parallel approaches for solving the problem, looking both at the initially defined frequent item set mining problem and at its extension to the sequence and graph mining domains. We identified three areas as key challenges to parallel algorithmic design in the context of frequent pattern mining: memory scalability, work partitioning, and load balancing.

REFERENCES

[1] Cheikh Tidiane Dieng, Tao-Yuan Jen, Dominique Laurent, and Nicolas Spyratos.Mining frequent conjunctive queries using functional and inclusion dependencies. VLDB J., 22(2):125 150, 2013.

[2] Bart Goethals, Dominique Laurent, Wim Le Page, and Cheikh Tidiane Dieng. Mining frequent conjunctive queries in relational databases through dependency discovery. Knowl. Inf. Syst., 33(3):655 684, 2012.

[3] Cheikh Tidiane Dieng, Tao-Yuan Jen, and Dominique Laurent. An effcientcomputation of frequent queries in a star schema. In Database and Expert Systems Applications, 21th International Conference, DEXA 2010, Bilbao, Spain, August 30 - September 3, 2010, Proceedings, Part II, pages 225 239, 2010.

[4] Kun He, Yiwei Sun, David Bindel, John E. Hopcroft, and Yixuan Li. Detecting overlapping communities from local spectral subspaces. In *2015 IEEE International Conference on Data Mining (ICDM 2015), Atlantic City, NJ, USA*, pages 769-774, 2015.

[5] Ayad Ibrahim, Hai Jin, Ali A. Yassin, and Deqing Zou. Towards privacy preserving mining over distributed cloud databases. In *Proceedings of the 2nd International Conference on Cloud and Green Computing (CGC 2012), Xiangtan, Hunan, China*, pages 130{136. IEEE Computer Society, 2012.

[6] Leila Ismail and Liren Zhang. Modeling and performance analysis to predict the behavior of a divisible load application in a cloud computing environment. *Algorithms*, 5(2):289{303, 2012.

[7] Fan Jiang and Carson Kai-Sang Leung. Stream mining of frequent patterns from delayed batches of uncertain data. In *Proceedings of the 15th International Conference on Data Warehousing and Knowledge Discovery (DaWaK 2013), Prague, Czech Republic*, pages 209{221. Springer-Verlag New York, Inc., 2013.

[8] Alfredo Cuzzocrea, Ladjel Bellatreche, and Il-Yeol Song. Data warehousing and olap over big data: Current challenges and future research directions. In *Proceedings of the 16th International Workshop on Data Ware-housing and OLAP (DOLAP 2013), San Francisco, California, USA*, pages 67{70. ACM, 2013.

[9] Malu Castellanos, Chetan Gupta, Song Wang, and Umeshwar Dayal. Leveraging web streams for contractual situational awareness in operational BI. In *Proceedings of the 2010 International Conference on Extending Database Technology/International*

Conference on Database Theory (EDBT/ICDT 2010) Workshops, Lausanne, Switzerland, pages7:18. ACM, 2010.

[10] Alfredo Cuzzocrea, Carson Kai-Sang Leung, and Richard Kyle MacKinnon. Mining constrained frequent itemsets from distributed uncertain data. *Future Generation Computer Systems*, 37:117{126, 2014.

[11] Alfredo Cuzzocrea, Domenico Saccia, and Jefrerey D. Ullman. Big data: A research agenda. In *Proceedings of the 17th International Database Engineering & Applications Symposium (IDEAS 2013), Barcelona, Spain*,pages 198{203. ACM, 2013.

[12] Alfredo Cuzzocrea. CAMS: OLAPing multidimensional data streams efficiently. In *Proceedings of the 11th International Conference on Data Warehousing and Knowledge Discovery (DaWaK 2009), Linz, Austria*,pages 48-62. Springer verlag, 2009.

[13] Yifan Chen, Xiang Zhao, Xuemin Lin, and Yang Wang. Towards frequent subgraph mining on single large uncertain graphs. In *2015 IEEE International Conference on Data Mining (ICDM 2015), Atlantic City,NJ, USA*, pages 41{50, 2015.

[14] Jeferey Dean and Sanjay Ghemawat. Mapreduce: simplified data processing on large clusters. *Commun. ACM*, 51(1):107{113, 2008.

[15] [dson Dela Cruz, Carson Kai-Sang Leung, and Fan Jiang. Mining `following' patterns from big sparse social networks. In *Proceedings of the International Symposium on Foundations and Applications of Big Data Analytics (FAB 2016), San Francisco, CA, USA*, pages 923{930. ACM,2016.

[16] Mohammad El-Hajj and Osmar R. Zaiane. Parallel bifold: Largescale parallel pattern mining with constraints. *Distributed and Parallel Databases*, 20(3):225{243, 2006.

[17] Mohammad El-Hajj and Osmar R. Parallel leap: Large-scale maximal pattern mining in a distributed environment. In *Proceedings of the 12th International Conference on Parallel and Distributed Systems (ICPADS 2006), Minneapolis, USA*, pages 135{142. IEEE, 2006.

[18] Fan Jiang, Carson Kai-Sang Leung, Dacheng Liu, and Aaron M. Peddle. Discovery of really popular friends from social networks. In *Proceed- ings of the 4th IEEE International Conference on Big Data and Cloud Computing (BDCloud 2014), Sydney, Australia*, pages 342{349, 2014.

[19] Dongme Sun, Shaohua Teng, Wei Zhang, Haibin Zhu, “An Algorithm to Improve the Effectiveness of Apriori”, In Proc. Int‟l Conf. on 6th IEEE Int‟l Conf. on Cognitive Informatics (ICCI'07), 2007.

[20] Mannila, H. and Toivonen, H.,”Discovering generalized episodes using minimal occurrences”, In Proc. of ACM Conference on Knowledge Discovery and Data Mining (SIGKDD), Pages 146–151, 1996.

[21] Anandhavalli M, Suraj Kumar Sudhanshu, Ayush Kumar and Ghose M.K. (2009) Optimized Association Rule Mining using Genetic Algorithm, Advances in Information Mining, ISSN:0975-3265, Volume 1, Issue 2, 2009, pp-01-04.

[22] Markus Hegland. The Apriori Algorithm – a Tutorial, CMA, Australian National University, WSPC/Lecture Notes Series, 22-27, March 30, 2005.

[23] Bart Goethals, Dominique Laurent, Wim Le Page, and Cheikh Tidiane Dieng. Mining frequent conjunctive queries in relational databases through dependency discovery. Knowl. Inf. Syst., 33(3):655 684, 2012.

[24] Woo, J., Xu, Y, “Market Basket Analysis Algorithm with Map/Reduce of Cloud Computing”, In Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications, 2001.

[25] Lin, Ming-Yen, Pei-Yu Lee, Sue-Chen Hsueh, "Apriori-based Frequent Itemset Mining Algorithms on MapReduce", In Proceedings of the 6th International Conference on Ubiquitous Information Management and Communication, ACM, 2012.