IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 3, March 2017

Data Mining Tool for SMBs with FIM Method

Prof. S.S. Pawar[1], Rupesh Wankhede[1], Hitesh Godse[1], Amar Puranik[1], Akash Surwase[1]

Department of Computer Engineering, Sinhgad College of Engineering, Pune, India[1]

Abstract: Data analytics is the science of examining raw data with the purpose of drawing conclusions about that information. Using these conclusions, retailers can make decisions that can significantly increase their revenue. However, small to medium sized businesses (SMBs) often cannot benefit from the advantages of data analysis due to lack of awareness, being technologically incompetent and other reasons. We propose to solve the above problem by developing a data analysis tool. The tool will run on a single computer- the same computer on which the source transactional data will be generated. It will be easy to use for those who are not comfortable with computers by providing a user-friendly interface and by displaying conclusions in an easy to read manner. These conclusions will guide the business owner (user) in making better business decisions. The goal behind creating the tool is to let the users realize, first-hand, the benefits data analysis can bring to their business. We will also be making use of an improved frequent-itemset mining (FIM) approach which will reduce the execution times of the tool.

Keywords: Data analysis, data mining, Apriori algorithm, Hadoop cluster.

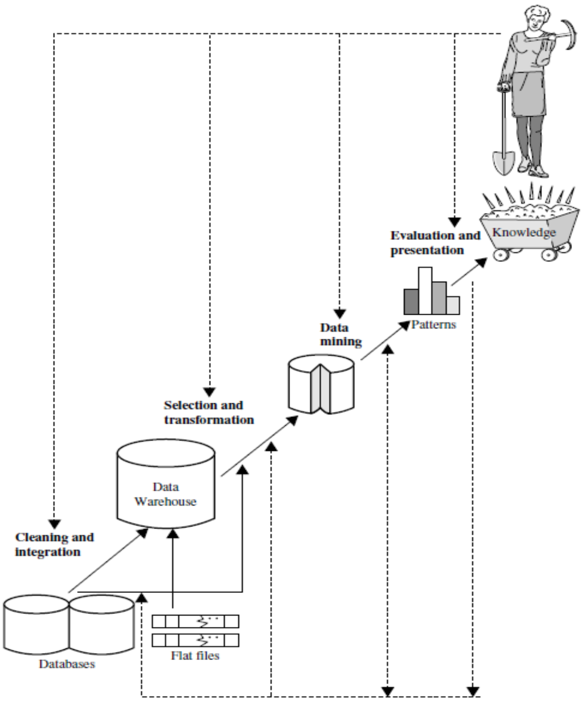
I. INTRODUCTION

Data analysis is a process of inspecting, cleaning, transforming, and modeling data with the goal of discovering useful information, suggesting conclusions, and supporting decision-making [1]. Data mining is a particular data analysis technique that focuses on modeling and knowledge discovery for predictive rather than purely descriptive purposes [1]. Our tool performs seven different types of analysis on the transactional data provided to it. The conclusions obtained from these analyses can help make better business decisions which can thus lead to a significant increase in sales. We will also be making use of a mining technique which will be parallelized by implementing it in Hadoop.

II. DATA ANALYSIS AND DATA MINING

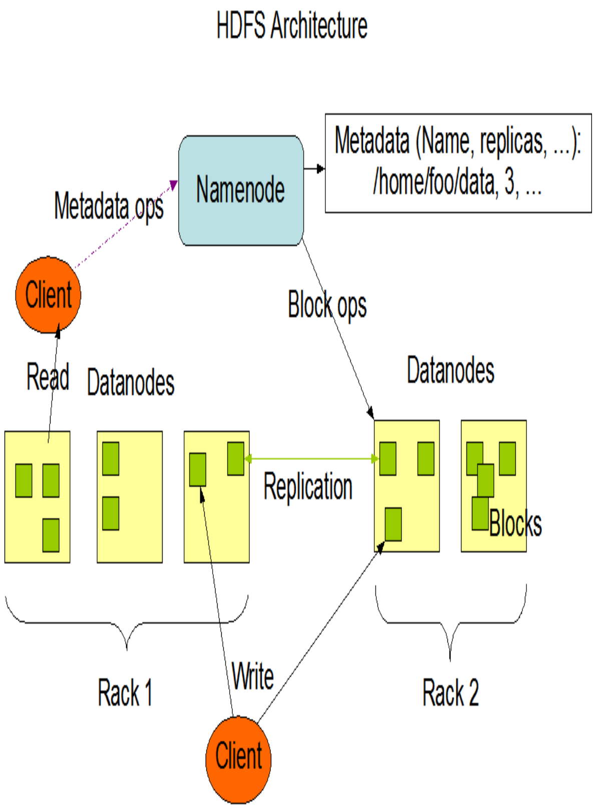
Data analytics (DA) is the process of examining data sets in order to draw conclusions about the information they contain, increasingly with the aid of specialized systems and software [2]. Data mining can be defined as the process of discovering interesting patterns and knowledge from large amounts of data [3]. Many people treat data mining as a synonym for a popularly used term, knowledge discovery from data, or KDD, while others view data mining as merely an essential step in the process of knowledge discovery [3].

The KDD process includes Data cleaning, data integration, data selection, data transformation, data mining, Pattern evaluation, knowledge presentation steps. Transactional data generated in a business can be analysed or mined to obtain insights about customers' shopping habits. These insights can help make better business decisions which can thus lead to a significant increase in sales. Data mining as a step in the process of knowledge discovery can be shown as below.



III.TECHNOLOGY

Hadoop is an open source, Java-based programming framework that supports the processing and storage of extremely large data sets in a distributed computing environment [4]. It is part of the Apache project sponsored by the Apache Software Foundation [4]. Hadoop makes it possible to run applications on systems with thousands of commodity hardware nodes, and to handle thousands of terabytes of data [4]. Its distributed file system facilitates rapid data transfer rates among nodes and allows the system to continue operating in case of a node failure [4]. This approach lowers the risk of catastrophic system failure and unexpected data loss, even if a significant number of nodes become inoperative [4]. Consequently, Hadoop quickly emerged as a foundation for big data processing tasks, such as scientific analytics, business and sales planning, and processing enormous volumes of sensor data, including from internet of things sensors [4]. As a software framework, Hadoop is composed of numerous functional modules [4]. At a minimum, Hadoop uses Hadoop Common as a kernel to provide the framework's essential libraries [4]. Other components include Hadoop Distributed File System (HDFS), which is capable of storing data across thousands of commodity servers to achieve high bandwidth between nodes; Hadoop Yet Another Resource Negotiator (YARN), which provides resource management and scheduling for user applications; and Hadoop MapReduce, which provides the programming model used to tackle large distributed data processing -- mapping data and reducing it to a result [4]. The Hadoop Distributed File System (HDFS) is the primary storage system used by Hadoop applications [5]. HDFS is a distributed file system that provides high-performance access to data across Hadoop clusters [5]. Like other Hadoop-related technologies, HDFS has become a key tool for managing pools of big data and supporting big data analytics applications [5].



When HDFS takes in data, it breaks the information down into separate pieces and distributes them to different nodes in a cluster, allowing for parallel processing [5]. The file system also copies each piece of data multiple times and distributes the copies to individual nodes, placing at least one copy on a different server rack than the others [5]. As a result, the data on nodes that crash can be found elsewhere within a cluster, which allows processing to continue while the failure is resolved [5]. HDFS is built to support applications with large data sets, including individual files that reach into the terabytes [5]. It uses master/slave architecture, with each cluster consisting of a single NameNode that manages file system operations and supporting DataNodes that manage data storage on individual compute nodes [5].

IV. MAPREDUCE

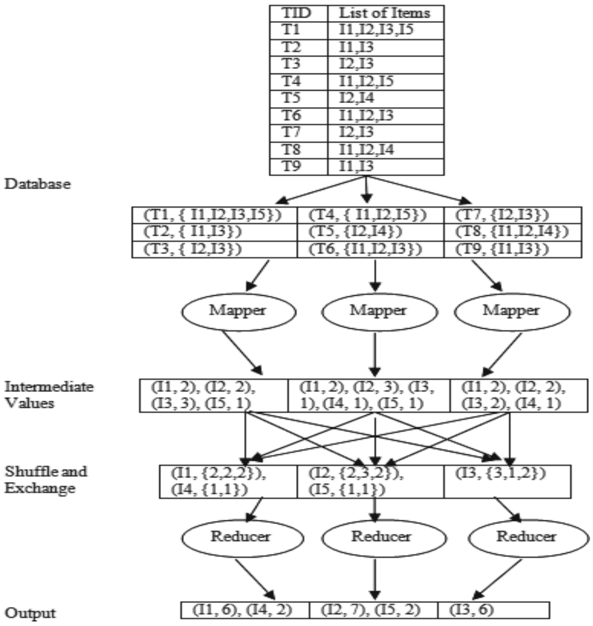
Hadoop MapReduce is a software framework for easily writing applications which process vast amounts of data (multi-terabyte data-sets) in-parallel on large clusters (thousands of nodes) of commodity hardware in a reliable, fault-tolerant manner [7]. A MapReduce job usually splits the input data-set into independent chunks which are processed by the map tasks in a completely parallel manner [7]. The framework sorts the outputs of the maps, which are then input to the reduce tasks [7]. Typically both the input and the output of the job are stored in a file-system [7]. The framework takes care of scheduling tasks, monitoring them and re-executes the failed tasks [7]. Typically the compute nodes and the storage nodes are the same, that is, the MapReduce framework and the Hadoop Distributed File System are running on the same set of nodes [7]. This configuration allows the framework to effectively schedule tasks on the nodes where data is already present, resulting in very high aggregate bandwidth across the cluster [7]. The MapReduce framework consists of a single master JobTracker and one slave TaskTracker per cluster-node [7]. The master is responsible for scheduling the jobs' component tasks on the slaves, monitoring them and re-executing the failed tasks [7]. The slaves execute the tasks as directed by the master [7].

V. APRIORI ALGORITHM ON HADOOP MAPREDUCE

Apriori algorithm is an iterative process and its two main components are candidate itemsets generation and frequent itemsets generation [9]. In each scan of database, mappers generate local candidates and reducers sum up the local count and result frequent itemsets [9]. The first step of the algorithm is to generate frequent 1-itemsets L_1 which is illustrated in Figure by an example. HDFS breaks the transactional database into blocks and distributes them to all mappers running on machines [9]. Each transaction is converted to a (key, value) pair where key is the TID and value is the list of items i.e. transaction. Each mapper reads one transaction at a time and outputs (key', value') pairs where key' is each item in transaction and value' is 1. The combiner combines the pairs with same key' and makes the local sum of the values for each key' [9]. The

DOI 10.17148/IJARCCE.2017.6336

output pairs of all combiners are shuffled & exchanged to make the list of values associated with the same key', as (key', list (value")) pairs [9]. Reducers take these pairs and sum up the values of respective keys [9]. Reducers output (key', value"') pairs where key' is an item and value"' is the support count ≥ minimum support, of that item [10,11,12]. Finally, the set of frequent 1-itemsets, L_1, is obtained by merging the output of all reducers [9].



TID	List of Items
T1	I1,I2,I3,I5
T2	I1,I3
T3	I2,I3
T4	I1,I2,I5
T5	I2,I4
T6	I1,I2,I3
T7	I2,I3
T8	I1,I2,I4
T9	I1,I3

FIG: Generation of frequent 1-itemsets. [9].

VI . FEATURES

The types of analysis our tool provides and the working behind them are given below:

1. Ad-hoc Query Analysis
2. Visualization of Data
3. Recommendation of Items
4. Recommendation of Item Assortments (Bundles)
5. Suggestion of Optimum Prices for Items
6. Identification of Seasonal Items
7. Identification of Brands Favored by Customers.

In ad-hoc query analysis, the user will be able to find specific data immediately. The user can find data according to some preset options. In Visualization, Data selected by the user will be displayed in different graphical representations which will help the user understand the data better. This will be achieved through the use of external APIs like Chart2D and JFreeChart. Using Recommendation of Items, the user(business owner) can identify the items which can be recommended to the customer for purchasing. The items to be recommended can be found out from the items the customer already has in his/her shopping cart using association rules.Item assortments will be packages of items that are frequently bought together. These items will be identified from frequent itemsets. The optimum price of an item is the price that will generate maximum profit. Seasonal items are the items that sell well during a particular time of the year. These can be found out by using ad-hoc queries in the backend. The highest selling items in a given time period derived using ad-hoc queries will be the seasonal items of that time period. We will be providing the user options consisting of different seasons and festivals in our tool. In Identification of Brands analysis, brands that are liked, trusted and bought by customers more often are identified and displayed to the user.

VII .CONCLUSION

To introduce data analysis and its benefits to small to medium sized businesses (SMBs), we developed an easy to use and effective data mining tool. We implemented it in the form of a web application with the data being stored in a Hadoop cluster. The tool makes use of a parallel frequent itemset mining (FIM) technique which was implemented using the MapReduce programming model. Since the mining is carried out in a parallel fashion, the mining performance of our tool is substantially better than that of sequential mining methods.

REFERENCES

1) HTTPS://EN.WIKIPEDIA.ORG/WIKI/DATA_ANALYSIS.
2) http://searchdatamanagement.techtarget.com/definition/data-analytics.
3) Data Mining: Concepts and Techniques, 3rd Edition. Jiawei Han, Micheline Kamber, Jian Pei.
4) http://searchcloudcomputing.techtarget.com/definition/Hadoop.
5) http://searchbusinessanalytics.techtarget.com/definition/Hadoop-Distributed-File-System-HDFS.
6) https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html.
7) https://hadoop.apache.org/docs/r1.2.1/mapred_tutorial.html.
8) https://cs.calvin.edu/courses/cs/374/exercises/12/lab/.
9) Review of Apriori Based Algorithms on MapReduce Framework, conference paper (June 2014), Sudhakar Singh, Rakhi Garg, P K Mishra.
10) M-Y. Lin, P-Y. Lee and S-C. Hsueh, "Apriori-based Frequent Itemset Mining Algorithms on MapReduce," in Proceedings 6th International Conference on Ubiquitous Information Management and Communication (ICUIMC '12), ACM, New York, Article 76, (2012).
11) N. Li, L. Zeng, Q. He and Z. Shi, "Parallel Implementation of Apriori Algorithm based on MapReduce," in Proceedings 13th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel & Distributed Computing, IEEE, pp. 236–241, (2012).
12) L. Li and M. Zhang, "The Strategy of Mining Association Rule Based on Cloud Computing," in Proceedings IEEE International Conference on Business Computing and Global Informatization (BCGIN), pp. 29–31, (2011).

DOI 10.17148/IJARCCE.2017.6336