ISSN 2348 – 9928

IJAICT Volume 2, Issue 1, January 2015 Doi:01.0401/ijaict.2014.02.04 Published Online 05 (1) 2015

EFFICIENT DATA ANALYSIS SCHEME FOR INCREASING PERFORMANCE IN BIG DATA

Mr. V. Vivekanandan
Computer Science and Engineering,
SriGuru Institute of Technology,
Coimbatore, Tamilnadu, India.

Ms. N. Karpagavalli
Computer Science and Engineering,
SriGuru Institute of Technology,
Coimbatore, Tamilnadu, India.

Abstract—***Big data is the process of handling large datasets. In today's scenario, data is growing exponentially faster than ever so the concept of Big data has emerged. It can perform data storage, data analysis, and data processing as well as data management techniques in parallel. The aim of this project is to use the classification technique before mapping the tasks into the resources. Usually, the MapReduce will take more time to decide the resource for performing the tasks which is to be allocated. Parallel Database technology is used to increase the performance of Big data because it allocate the tasks in parallel into the resources. In this model, for classifying the tasks, Ensemble Classifier is used. Along with Ensemble Classifier, Map Reduce model and Parallel Database Technology is associated which increases the efficiency and throughput of Big Data by reducing the scheduling time.***

Keywords— ***Map Reduce, Hadoop, Ensemble Classifier, Parallel Database***

I. INTRODUCTION

Big data is capable of handling large datasets at a time. It can perform data storage, data analysis, and data processing and data management techniques in parallel. Big data can process several peta bytes (10^{15}) of data in seconds. It can handle both structured and unstructured data at a time. Big data spends 70% of the time on gathering and retrieving the data and remaining 30% of the time is spend on analyzing the data. Big data can process even several peta bytes of data in seconds. Big data analytics will be most useful for hospital management and government sectors especially in climate condition monitoring. There are three characteristics of big data namely volume, velocity and variety. The characteristics are explained in detail below.

- ***Volume***

Many factors contribute to the increase in data volume. Volume refers to the sense of storage in Big data. For example, in facebook 2.5 peta bytes of data are processed per day. Through the internet 2.5 Quintillion (10^{27}) bytes of data are processed per day. In order to store those large amounts of data we use Big data.

- ***Velocity***

Data is streaming in at unprecedented speed and must be dealt with in a timely manner. Velocity refers to the speed and performance of Big data. For example, internet can process only 4-5 Mb of data per second but in Big data 10Mb of data can be processed per second.

- ***Variety***

Data today comes in all types of formats. Variety refers to the types of data used in Big data. Structured data refers to numeric data in traditional databases. Unstructured data like text documents, email, video, audio, stock ticker data and financial transactions. For example, in Data warehousing and Data Mining either structured or unstructured data can be used. But in Big data both structured and unstructured data can be used at a time.

1.1 Hadoop

Hadoop is the most popular open source framework used in Big data to handle large datasets. It is a batch oriented system. Hadoop is used to analyze user interaction data. It is linear scalable on low cost commodity hardware. It is designed to parallelize data processing across computing nodes to speed computations and hide latency.

- ***Hadoop Architecture***

The architecture of Hadoop consists of one master node and many slave nodes. In the master node there will be a MapReduce model which is used for computation purpose and a Hadoop Distributed File System (HDFS) which is used to store large amount of data. Also in each slave node there will be a Map Reduce as well as HDFS. In the Map Reduce, the master node will take care of allocating the tasks to the slave nodes for processing. The HDFS in the master node will allocate the storage space to each slave node and also keeps track of where each data is located. Once the slave node finishes the given tasks it will send the results back to the master node. There will be commodity hardware in the Hadoop architecture which is used to improve the system's performance.

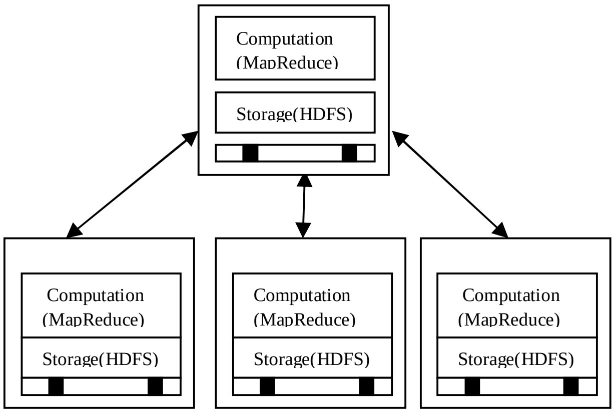

Fig 1: Architecture of Hadoop

1.2 HDFS

HDFS is the storage component of Hadoop and is based on Google's Google File System (GFS). It is optimized for high throughput and works best when reading and writing large files. The blocks are replicated to nodes throughout the cluster based on the replication factor (default is 3). Replication increases reliability and performance. The architecture of HDFS consists of three daemons: They are:

- NameNode (Master)
- Secondary NameNode (Master)
- DataNode (Slave)

- ***NameNode***

The namenode stores all the metadata. The namenode contains information about file locations in HDFS, information about file ownership and permissions, name of the individual blocks and the location of the blocks. Metadata is stored on disk and it is read when the namenode daemon starts up. Any changes to metadata are made in RAM and the changes are also written to log file on the disk called edits. There can be only one namenode in the Hadoop cluster.

- ***Secondary NameNode***

Secondary name node performs memory-intensive administrative functions for the namenode. Namenode keeps information about all the files and blocks. Namenode writes metadata changes to an editing. Secondary namenode periodically combines a prior file system snapshot and editing into a new snapshot. New snapshot is transmitted back to namenode. Secondary namenode should not run on a separate machine in a large installation as it requires as much RAM as namenode.

- ***DataNode***

Datanodes contains the actual contents of the files which are stored as blocks on the slave nodes. Blocks are simply files on the slave nodes underlying filesystem. Nothing on the slave node provides information about what underlying file the block is a part of. All those informations are only stored in the namenode. Datanodes must send block reports to both namenodes since the block mappings are stored in a namenode's memory, and not on disk.

1.3 MapReduce

MapReduce was introduced by Google in 2004 for executing set of functions against large amount of data. It is a software framework for processing large datasets in a distributed fashion over several machines. The core idea behind MapReduce is mapping the dataset into collection of key/value pair and then reducing all pairs with same key.

- ***Map***

The map step acts as a master node which takes the input and split the work to all the slave nodes. Then each slave node processes the result for the given input and passes the result back to the master node.

- ***Reduce Step***

The master node collects the results for the given input from all the slave nodes and combines the results to produce output.

II. LITERATURE SURVEY

In the literature support, the various research papers are surveyed to find the problems in Big data and also to find the solutions for solving those problems. In data mining and data warehousing , 95% of the time is spend on gathering and retrieving the data and only 5% of the time is spend on analyzing the data. Data mining and data warehousing cannot process large amount of data in parallel. It analyses the data by using application software. Multidimensional database system is used for storing and managing the data in Data mining and data warehousing. Data mining is a single technology which applies many older computational techniques from statistics, machine learning and pattern recognition. To overcome the above problems we use Big Data. Big Data processes large amount of data by using MapReduce Technology and Parallel Database Technology is combined with MapReduce in order to perform parallel computation. The following research papers are discussed below.

TABLE I- Review of Big Data

S.No	Approaches	Functionalities
1	Exploration on Big Data Oriented Data Analyzing and Processing Technology[7]	Big Data integrates storage, analysis, management, processing and application together in parallel with the help of MapReduce and Parallel Database Technology so that the efficiency of large amount of data is improved
2	Algorithm and Approaches to Handle Large Data-A Survey[2]	Inorder to improve efficiency and performance of computation we combine genetic algorithm and decision tree algorithm which lead to less time and space complexity so that the efficiency and performance of computation is improved.
3	MapReduce: Simplified Data Processing on Large Clusters[3]	In order to simplify the processing of data on large clusters, inter-machine communication has been used to make the efficiency high and also to minimize the time consumption so that large datasets can be processed quickly.
4	Dynamo: Amazon's Highly Available Key-Value Store[4]	MapReduce programming model uses the concept of key/value pair to perform computation for large amount of data because Key-value storage system, data versioning and partitioning algorithm is used to provide reliability which highly increases scalability, reliability and durability when large amount of data is used.
5	Knowledge Mining in Supervised and unsupervised Assessment Data of Students Performance[1]	The overall performance of the students are assessed using data mining techniques such as association rule mining technique and apriori algorithm in which the overall idea of the institution, curriculam, academics are clearly observed.
6	A Library of Constructive Skeleton for Sequential style of Parallel Programming[5]	Constructive algorithm is used in sequential style, but in parallel programming model skeleton library, SkeTo is used. It has high description power which can be extended in a compatible way and also supports sequential style of parallel programming.
7	Compression, Clustering and Pattern Discovery in Very High- Dimensional Discrete-Attribute Data Sets[6]	PROXIMUS framework is used for reducing large datasets into smaller datasets. PROXIMUS use both sub sampling and compression of data before applying computationally expensive algorithms. It improves performance and scalability by means of algebraic technique and data structure.
8	N. Beckmann, H. -P. Kriegal, R. Schneider, and B.Seeger, "The R*-Tree: An Efficient and Robust AccessMethod for Points and Rectangles," Proc. ACMSIGMOD, May 1990.[8]	The R*-tree algorithm is very attractive because it efficiently supports the point and spatial data in parallel and the implementation cost is slightly higher than other R-tree algorithm .The search time is $O(3^d)$.
9	S. Arya, D. Mount, N. Netanyahu, R. Silverman, A. Wu,"An Optimal Algorithm for Approximate Nearest Neighbor Searching in Fixed Dimensions, " Proc. Fifth Symp. Discrete Algorithm (SODA), 1994, pp. 573	The nearest neighbor technique is implemented in which searching an object is expensive in high dimensional data space. It is highly time consuming. The search time is O(dn log n).
10	Lawrence 0. Hall, NiteshChawla, Kevin W. Bowyer,"Decision Tree Learning on Very Large Data Sets",IEEE, Oct 1998[10]	The decision tree algorithm is implemented which is simple, fast and produce accurate results. In this approach the rules for an agent is generated from a large training set.
11	Zhiwei Fu, Fannie Mae, "A Computational Study of Using Genetic Algorithms to Develop Intelligent Decision Trees", Proceedings of the 2001 IEEE congress on evolutionary computation, 2001.	High quality decision tree is produced by dividing the data set into training, scoring and test sets. Local greedy search is used throughout the dataset so the consumption of the search time is less.

12	Mr. D. V. Patil, Prof. Dr. R. S. Bichkar, "A Hybrid Evolutionary Approach To Construct Optimal Decision Trees with Large Data Sets", IEEE, 2006	Decision tree learning with optimization is used inorder to handle large datasets. As optimization is used the performance of the classification is improved and the size of the tree is greatly reduced with high accuracy.
13	Yen-ling Lu, chin-shyurngfahn, "Hierarchical Artificial Neural Networks For Recognizing High Similar Large Data Sets. ", Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, August 2007.	When neural network is used with large datasets the accuracy rate is very high. The search time is less and the performance is greatly improved.

III. PROPOSED SYSTEM

The classification technique is used to classify the whole dataset before mapping the tasks into the resources so that it reduce the time span, whereas during later period each and every data of whole dataset were analyzed individually and then mapped into the resources which consumes more time to complete the task. To classify and analyze the data before mapping, an Ensemble Classifier is used. To increase the efficiency and throughput of Big data we make use of MapReduce and Parallel Database technology.

Earlier each and every data were analyzed individually, so it takes more time to decide to which resource the tasks has to be allocated. In the MapReduce model, the map step will map the tasks into the resources based on the key/value pair to perform computation and the reduce step will aggregate all the results from the map step and finally produce a single output. The Parallel Database Technology is used to perform the computation of large data in parallel which also improves the performance of Big data. The input in this project will be the large dataset in which the whole dataset will be classified and analyzed before mapping the tasks into the resources by reducing the time span to analyze the data.

For classifying the tasks, Ensemble Classifier is used. An Ensemble Classifier is the group of different classifiers which make the classifiers to process in parallel and also shares the knowledge of fastest processing classifier to others. Therefore, the data's will be processed with minimal scheduling time (the map class will not take time to decide to which resource the task has to be allocated).

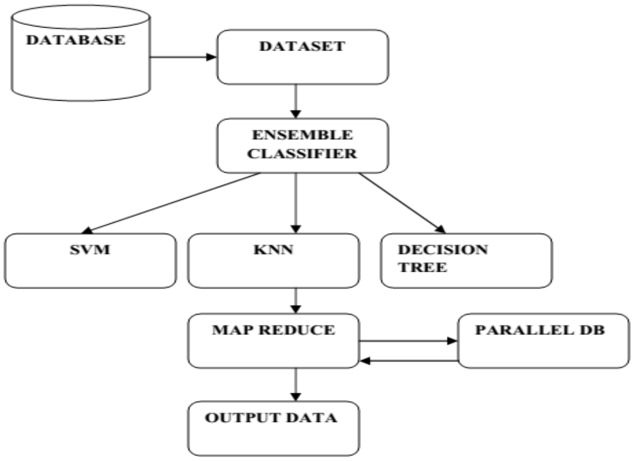

Fig 2 : Proposed Architecture

3.1 Ensemble Classifier

An Ensemble Classifier is a group of different classifiers in which the classifiers will be made to perform in parallel and the knowledge of the fastest processing classifier will be shared to other classifiers. An Ensemble classifier will give more accurate results when compared to other individual classifiers. The dataset will be loaded into an Ensemble Classifier. There are three classifiers used namely Support Vector Machine, K-Nearest Neighbor and Decision Tree.

SVM are supervised learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. The basic SVM takes a set of input data and predicts, for each given input, which of two possible classes forms the output, making it a non-probabilistic binary linear classifier. The SVM checks the incoming dataset and verifies whether the data is knowledgeable data or not. If the incoming data is a knowledgeable data then the support vector machine will support the incoming data to proceed for processing. If the incoming data is a new data then that data will be analyzed by the SVM. After analyzing the new data that data can be used for further processing.

Decision Tree Classifier is a simple and widely used classification technique. It applies a straightforward idea to solve the classification problem. Decision Tree Classifier poses a series of carefully crafted questions about the attributes of the test record. Each time it receives an answer, a follow-up question is asked until a conclusion about the class label of the record is reached. The Decision Tree classifier will look for the incoming dataset and will split the dataset based on the category wise. It splits the attributes in the dataset. The attribute which has the highest information gain will be chosen as a splitting attribute.

The K-NN is a non-parametric method for classification and regression that predicts objects' "values" or class memberships based on the *k* closest training examples in the feature space. K-NN is a type of instance-based learning, or lazy learning where the function is only approximated locally and all computation is deferred until classification. The *k*-nearest neighbor algorithm is amongst the simplest of all machine learning algorithms: an object is classified by a majority vote of its neighbors, with the object being assigned to the class most common amongst its *k* nearest neighbors (*k* is a positive integer, typically small). If *k* = 1, then the object is simply assigned to the class of that single nearest neighbor. The incoming dataset will be analyzed by K-NN and if the data is similar to the data present in the nearest neighbor then that data will be accepted and processed. If the incoming data is not similar to the data present in nearest neighbor then that data will be analyzed for further processing.

3.2 MapReduce and Parallel Database Technology

An efficient data analysis framework is constructed using MapReduce programming model and Parallel Database Technology. MapReduce programming model is established to map the incoming tasks into the resources and also greatly reduces the workloads of the resources. Parallel Database Technology is used for the processing of the tasks to be done in parallel by utilizing the resources efficiently, thereby increasing the performance of Big data.

The Map technology mainly processes a group of input data record and distributes data to several servers and operation systems. Its means of processing data is a strategy based on the key/value pair. The Reduce technology mainly occupies itself in summarizing and processing the result after processing the above key/value. Map Reduce is designed for mass composed of low-end computer cluster, its excellent scalability has been fully verified in industry. Map Reduce has low requirement to hardware. Map Reduce can store data in any format, can achieve a variety of complex data processing function. Analysis based on the Map Reduce platform, without the need of complex data preprocessing and writing in the database process.

Parallel computing includes two aspects: data parallel processing and task parallel processing. In terms of the data parallel processing means, a large-scale task to be solved can be dissembled into various system sub-tasks with the same scale and then each sub-task will be processed. As such, compared to the whole task, it will be easy to process. Adopting the task paralleling processing mode might cause the disposal of tasks and coordination of relationships overly complicated. Using the parallel database technology is a means for realizing the parallel processing of data information. Parallel database support standard SQL language, through the SQL to provide data access service, SQL is widely used because it is simple and easy to apply. But in big data analysis, the SQL interface is facing great challenges. The advantage of SQL comes from packaging the underlying data access, but the packaging affects its openness to a certain extent. User-defined functions which provided by parallel database is mostly based on the design of a single database instance, and therefore they cannot be executed in parallel cluster, it means that the traditional way is not suitable for the processing and analysis of big data.

IV. CONCLUSION

In this paper, the study of MapReduce programming model is done to reduce the workloads on the resources and also to allocate the tasks into the resources. The Parallel Database Technology is used to perform the computation tasks in parallel which increases the performance of Big data. In order to reduce the scheduling time for allocating the tasks into resources, classification technique is used before MapReduce and Parallel Database technology. For classifying the tasks an Ensemble classifier is used. An Ensemble classifier is a group of different classifiers such as Support Vector Machine (SVM) classifier, Decision Tree classifier, K-Nearest Neighbor (KNN) classifier etc. The study of these various types of classifiers is done to share the knowledge of the fastest processing classifier to others which will greatly reduce the scheduling time. Along with Ensemble Classifier, MapReduce programming model and Parallel Database Technology is used to increase the efficiency and throughput of Big data.

References

[1] Anwar M. A. and Naseer Ahmed,“Knowledge Mining in Supervised and Unsupervised Assessment Data of Students’ Performance”, 2011 2nd International Conference on Networking and Information Technology.

[2] Chanchal Yadav, 2 Shuliang Wang, 3 Manoj Kumar 1 CSE, Amity University Noida, Uttar Pradesh, India, “Algorithm and Approaches to Handle Large Data - A Survey”, Volume 2, Issue 3, June 2013.

[3] Dean J, Ghemawat S. “MapReduce: Simplified data processing on large clusters”/ / Proceedings of the 6th Symposium on Operating System Design and Implementation(OSDΓ 04) .San Francisco, California, USA, 2004: 137-150.

[4] Giuseppe DeCandia, Deniz Hastorun, Madan Jampani, Gunavardhan Kakulapati, Avinash Lakshman, Alex Pilchin, Swaminathan Sivasubramanian, Peter Vosshall and Werner Vogels, “Dynamo: Amazon’s Highly Available Key-value Store”,SOSP’07, October 14-17, 2007.

[5] Mehmet Koyutu¨rk, Ananth Grama, and Naren Ramakrishnan, “Compression, Clustering, and Pattern Discovery in Very High-Dimensional Discrete-Attribute Data Sets”, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 17, NO. 4, APRIL 2005.

[6] N. Beckmann, H. -P. Kriegal, R. Schneider, and B.Seeger, “The R*-Tree: An Efficient and Robust Access Method for Points and Rectangles,” Proc. ACM SIGMOD, May 1990.

[7] S. Arya, D. Mount, N. Netanyahu, R. Silverman, A. Wu, “An Optimal Algorithm for Approximate Nearest Neighbor Searching in Fixed Dimensions, ” Proc. Fifth Symp. Discrete Algorithm (SODA), 1994, pp. 573-582.

[8] Zhiwei Fu, Fannie Mae, “A Computational Study of Using Genetic Algorithms to Develop Intelligent Decision Trees”, Proceedings of the 2001 IEEE congress on evolutionary computation, 2001.

[9] Mr. D. V. Patil, Prof. Dr. R. S. Bichkar, “A Hybrid Evolutionary Approach To Construct Optimal Decision Trees with Large Data Sets”, IEEE, 2006.

[10] Yen-ling Lu, chin-shyurngfahn, “Hierarchical Artificial Neural Networks For Recognizing High Similar Large Data Sets. ”, Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, August 2007.