A survey: classification of huge cloud Datasets with efficient Map - Reduce policy

Miss Apexa B. Kamdar[#1], Prof. Jay M. Jagani[*2]

[#]*M. E. Computer Engineering (Pursuing), Computer Engineering Department, Darshan Institute of Engineering and Technology, Rajkot, Gujarat, India.*

[*]*M. Tech. Computer Engineering, Assistant Professor of Computer Engineering Department, Darshan Institute of Engineering and Technology, Rajkot, Gujarat, India.*

***Abstract* - Cloud computing has become a feasible mainstream solution for data processing, storage and distribution. It assures on demand, scalable, pay-as-you-go compute and storage capacity. To analyze such huge data on clouds, it is very important to research data mining approach based on cloud computing model from both theoretical and practical views. There are large amount of data in cloud database or any other cloud file systems, then apply mining on that data to extract knowledge. Data mining is the process of analyzing data from different perspective and shortening it into useful information. For that Naïve Bayes and support vector machine algorithms are used, which are classification algorithms. In this paper both algorithms are used with MapReduce policy and get the high accuracy, efficiency, high performance. Hadoop is an open source implementation of Map Reduce which can achieve better performance.**

***Keywords* - Cloud computing, Data Mining, Naïve bayes, support vector machine, Hadoop, Map Reduce.**

I. INTRODUCTION OF CLOUD COMPUTING

Cloud computing is computing in which large groups of remote servers are networked to allow centralized data storage and online access to computer services or resources. . Cloud computing is a computing platform based on Internet, it provide the hardware and software resources on-demand to demanders through this platform. A cloud can be considered as a large group of interconnected computers which can be personal computers or network servers; they can be public or private. Clouds can be classified as public, private or hybrid. Everyday huge data are created so that demands for the advancement in data collection and storing technology. So that cloud computing is allow for store the data. Cloud computing is a computing platform based on Internet, it provide the hardware and software resources on-demand to demanders through this platform. A cloud can be considered as a large group of interconnected computers which can be personal computers or network servers; they can be public or

private. Cloud is responsible for maintaining and updating training data set for classification [9]. Cloud computing basically provides three different types of service based architectures are SaaS, PaaS, and IaaS. Architecture of cloud computing is shown in Fig. 1.

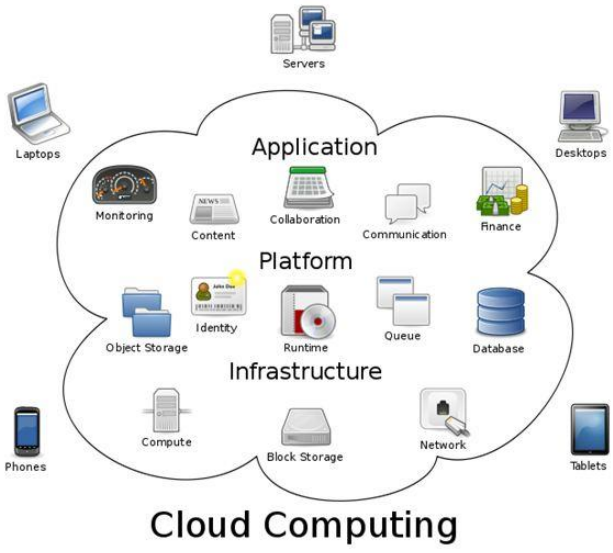

Fig. 1 Architecture of cloud computing

Cloud Mining can be considered as a new approach to apply Data Mining. There is a lot of data and regrettably this huge amount of data is difficult to mine and analyze in terms of computational resources. With the cloud computing paradigm the data mining and analysis can be more accessible and easy due to cost effective computational resources. Here we have discussed the usage of cloud computing platforms as a possible solution for mining and analyzing large amounts of data.

II. HADOOP AND MAP REDUCE

Hadoop is a distributed foundation infrastructure developed by the Apache. Users can easily develop and operate applications of mass data on the Hadoop; its core is HDFS, graphs and HBase. HDFS is open source. Hadoop is the most popular implementation of the Map Reduce programming

model. Hadoop is a Java-based programming framework that supports the processing of large data sets in a distributed computing environment. Hadoop can provide much needed robustness and scalability option to a distributed system as Hadoop provides inexpensive and reliable storage. The Hadoop distributed file system (HDFS) is a distributed, scalable, and moveable file-system written in Java for the Hadoop framework. HDFS is suitable for processing large data sets. HDFS is highly fault-tolerant and is designed to be arranged on low-cost hardware [7].

Map Reduce is a parallel data processing software framework for developing scalable applications and processing of huge amounts of data (multi-terabyte data-sets) in-parallel on large clusters (thousands of nodes) of commodity hardware in a consistent, fault-tolerant manner [1]. A Map Reduce job usually splits the input data-set into self-determining chunks which are processed by the map tasks in a completely parallel manner. The Map Reduce framework sorts the outputs of the maps, which are then specified as input to the reduce tasks. Both the input and output of the job are stored in the file system. The framework takes care of arrangement tasks, observing them and re-executes the failed tasks.

A. MAP Function

- Master node takes huge data input and split it into smaller sub problems, distributes these to worker nodes [4].
- Worker node may do this again; leads to a multi-level tree structure.
- Worker processes smaller data and give back to master.

B. REDUCE Function

- Master node takes the reply to the sub problems and combines them in a predefined way to get the output to original problem.

III. DATA MINING

Data mining is to extract useful knowledge from an existing dataset. Data mining is the process of analyzing data from different perspectives and summarizing it into useful information, information that can be used to increase profits, cuts costs, or both.

Classification: This category of algorithms deals with an unknown structure to a well known structure.

A. Naïve bayes algorithm

The Bayesian Classification represents a supervised learning method as well as a statistical method for classification. Naïve bayes is based on the Bayesian theorem [8]. This classification method analyses the relationship between each attribute and class for each case to derive a conditional probability for the relationships between the attribute values and the class. Naïve Bayesian classifiers must approximate the probabilities of a feature having a certain feature value. Naïve Bayes have also exhibited high accuracy and speed when applied to large databases. Naive Bayes classifiers often work much enhanced in many complex real-world situations than one might expect. This implementation is divided into training and prediction stages.

The training stages including three stages. First, the Input Format which is belonged to the Hadoop framework loads the input data into small data blocks known as data fragmentation, and the size of each data fragmentation is 5M, and the length of all of them is the same, and each split is divided into records. Second, the map function statistics the categories and properties of the input data, including the values of categories and properties and Third stage, the reduce function aggregates the number of each attribute and category value and then output the training model [2].

Prediction Stage, Predicate the data record with the output of the training model. Implementation of Naïve Bayes based on Map Reduce has very good performance and reduced the training time.

To demonstrate the concept of Naïve Bayes Classification, consider the knowledge of statistics. Let Y be the classification attribute and X{x1,x2,…,xk} be the vector valued array of input attributes, the classification problem simplifies to estimating the conditional probability P(Y | X) from a set of training patterns. P(Y | X) is the posterior probability, and P(Y) is the prior probability. Suppose that there are m classes, Y1, Y2… Ym. Given a tuple X, the classifier will predict that X belongs to the class having the highest posterior probability. The Naïve Bayes classifier predicts that tuple X belongs to the class Yi if and only if

$$P\ Yi|X\ >=\ P\ Yj|X \qquad (1)$$

The Bayes rule states that this probability can be expressed as the formulation

$$P(Yi|X) = \frac{P(X|Yi)P(Yi)}{P(X)} \qquad (2)$$

As P(X) is constant for all classes, only P(X | Yi) p(Yi) needs be maximized. The prior probabilities are estimated by the probability of Yi in the training set. In order to reduce computation in evaluating P(X|Yi), the Naïve Bayes assumption of class conditional self-rule is made. So the equation can be written into the form of

$$P(X|Yi) = \prod_{k=1}^{n} P(Xk|Yi) \qquad (3)$$

And we easily estimate the probabilities P(X1 | Yi), P(X2 | Yi),….,P(Xk | Yi), from the training tuples. The predicted

class label is the class Yi for which P(X | Yi) p(Yi) is the maximum.

B. Support Vector Machine Classification algorithm

SVM are typically used for learning classification, regression or ranking function. SVM are based on statistical learning theory and structure. SVM try to construct a separating hyperplane maximizing the margin between two data sets according to their classes [5].

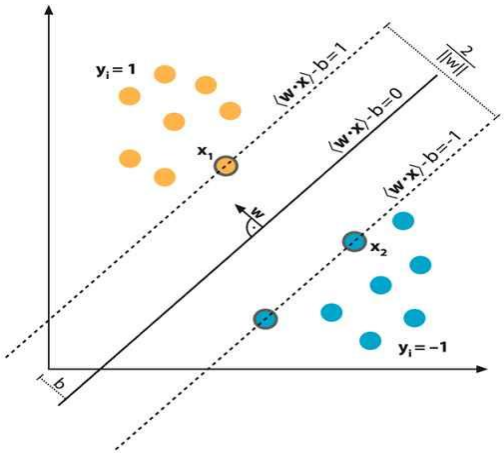

Fig. 2 Training data samples for two different classes

Hyperplane which has the largest distance to the neighboring data points of the both classes, because the larger the margin the better the generalization error of the classifier is getting and give the high accuracy [3].

SVM has been used in many pattern recognition and regression estimation problems and has been applied to the problems of dependency estimation, forecasting, and constructing intelligent machines. The idea behind the linear SVM algorithm is the search for the best hyperplane (called the optimal hyperplane) separating the data classes. This hyperplane can be described by a single linear equation.

$$w.Xi + b = 0 \qquad (4)$$

Where w (normal vector to the hyperplane) and b (offset) are calculated during SVM training. In the training stage, SVM tries to find the hyperplane as shown in Fig. 2 which minimizes the classification error while simultaneously maximizing the shortest distances from this hyperplane to the closest training samples of each class (d+ for class(+1), d- for class (-1)). The distance d+ and d-, equal to 1/w, define the margin associated with the separating hyperplane. Optimization of this margin is obtained by solving the constrained quadratic optimization problem.

$$\text{Minimize} \left(\|w\|^2 + c\sum_{i=1}^{n} \zeta i \right) \qquad (3.2)$$

$$\text{Subject to } \zeta i + yi \;\; w.Xi + b \;\; -1 \geq 0 \text{ with } \zeta i \geq 0$$

SVM tries to maximize the margin while keeping the classification error as low as possible. The classification error here is represented by the distance ζi of the misclassified sample i and the corresponding margin hyperplane. C called the regularization meta–parameter controls the trade-off between the two conflicting objectives: when C is small, margin maximization is emphasized whereas when C is large, error minimization is predominant.

IV. EXISTING WORK

A. Overview of Existing Work for implementation of the parallel Naïve Bayes

In traditional naïve bayes algorithm use with parallel implementation, the implementation of the parallel Naïve Bayes MapReduce model is divided into training and prediction stages.

1) Training Stage:

The distributed computing of Hadoop is divided into two phases which are called Map and Reduce. First, the Input Format which is belonged to the Hadoop framework loads the input data into small data blocks known as data fragmentation, and the size of each data fragmentation is 5M, and the length of all of them is equal, and each split is divided into records.

Map function statistics the categories and properties of the input data, including the values of categories and properties. The attributes and categories of the input records are separated by a comma, and the final attribute is the property of classification. Finally, the reduce function aggregates the number of each attribute and category value, which results in the form of (category, Index1:count1, Index2: count,…,Indexn: countn), and then output the training model. Its achievement is described as follows [2].

Algorithm Produce Training: map (key, value)
Input: the training dataset
Output: <key', value'> pair, where key' is the category, and value' the frequency of attribute value

1. FOR each sample DO BEGIN
2. Parse the category and the value of each attribute
3. count_thefrequence of the attributes
4. FOR each attribute value DO BEGIN
5. Take the label as key', and attribute index: the frequency of the attribute value as value'
6. Output<key', value'>
7. END
8. END

Algorithm Produce Training: reduce (key, value)
Input: the key and value output by map function
Output: <key', value'> pair, where key' is the label, and value' the result of frequency of attribute values

1. sum_0
2. FOR each attribute value DO BEGIN
3. sum+=value.next.get ()
4. END
5. Take key as key', and sum as value'
6. output<key', value'>

2) Prediction Stage:

Predicate the data record with the output of the training model. The implementation of the algorithm is stated as follows: first, use the statistical values of attribute values and category values to train the unlabeled record. In addition, use the distributed store to improve the efficiency of the algorithm in the procession of the algorithm implementation. Its implementation is described as follows [2].

Algorithm Produce Testing: map (key, value)
Input: the test dataset and the Naive Bayes Model
Output: the labels of the samples

1. modeltype_newModelType ()
2. categories_modeltype.getCategorys ()
3. FOR each attribute value not NULL DO BEGIN
4. Obtain one category from categories
5. END FOR
6. FOR each attribute value DO BEGIN
7. FOR each category value DO BEGIN
8. pct_counter(attribute, category)/Counter(category)
9. result_result*pct
10. END FOR
11. END FOR
12. Take the category of the max result as key' and the max result as value'
13. output<key', value'>

The comparing experiment shows that the performance of the improved algorithms is higher than the general methods with large data set. And this verifies the Bayesian algorithm runs on the cloud environment is more efficient than the traditional Bayesian algorithm [2]. However, due to the size of data sizes, feature, and the number of different categories, the time that the algorithm spent is not appear a linear relationship. Since running Hadoop jobs, start the cluster first which takes a little of time, so when the size of data set is minor, the data processing time is relatively longer. And this also verified the Hadoop is perfect to process huge amounts of data.

V. STEPS FOR NAÏVE BAYES AND SUPPORT VECTOR MACHINE COMBINE ALGORITHM

1) The proposed algorithm first initializes the weight of training examples to 1/n, where n is the total number of examples in training dataset, and then creates a new dataset from training dataset using selection with replacement technique.
2) After that it calculates the prior and conditional probabilities of new dataset, and classifies the training examples with these probabilities value.
3) The weights of the training examples updated according to how they were classified. If a training example is misclassified then its weight is increased, or if correctly classified then its weight is decreased.
4) Then the algorithm creates another new data set with the misclassification error produced by each training example from training dataset, and continues the process until all the training examples are correctly classified.
5) To classify a new example use all the probabilities in each round and consider the class of new example wit highest classifier's vote.
6) Then difference dataset provided to SVM for further classification.

VI. ADVANTAGES OF USING DATA MINING WITH CLOUD COMPUTING

Cloud computing combined with data mining can provide powerful capacities of storage and computing and an excellent resource management. It's give an efficient and high-performance computing is very necessary for a successful data mining application. Data mining in the cloud computing surroundings can be considered as the future of data mining because of the advantages of cloud computing paradigm. Cloud computing provides greater facility in data mining and data analytics. The major concern about data mining is that the space required by the operations and item sets is very large. But if we combine the data mining with cloud computing we can save a huge amount of space. This can benefit us to a great extent.

VII. SCOPE AND PURPOSE OF THE WORK

Hadoop being a popular parallel processing platform for data, many data mining algorithms are migrating towards Hadoop. In this work, migrating the data mining algorithms to Hadoop platform. Once these problems are identified Hadoop platform can be improved. These improvements will accelerate the performance of the data mining algorithms onto the Hadoop and it will attract more data mining operations to be moved to Hadoop platform.

VIII. CONCLUSION AND FUTURE WORK

When use of Naïve bayes classification algorithm in Hadoop then we can get the fast work and use of SVM algorithm then we can get the more accuracy so that apply both the algorithm we can get the high performance, reliability, more accuracy, faster execution compare to traditional algorithm. Use hadoop tool and run classification algorithm in it with the use of map reduce. Use naïve bayes

and SVM classification algorithm combine use and improve the accuracy, efficiency, less time require than Naïve Bayes algorithm.

ACKNOWLEDGMENT

I consider it my privilege to be able to take up the challenges and do some inward searching, self evaluation and explore the best. I also wish to express my heartfelt appreciation to my parents, my friends, my colleagues and many who have rendered their support for the successful works towards the completion of the research work, both explicitly and implicitly.

REFERENCES

[1] B. Liu, E. Blasch, Y. Chen, D. Shen, G. Chen, "Scalable Sentiment Classification for Big Data Analysis Using Naive Bayes Classifier", IEEE Intl Conf. on Big Data, Oct. 2013.

[2] Lijuan Zhou, Hui Wang, Wenbo Wang, "Parallel Implementation of Classification Algorithms Based on Cloud Computing Environment", TELKOMNIKA Indonesian Journal of Electrical Engineering, Vol.10, No.5, September 2012, pg. no. 1087-1092.

[3] Seyed Reza Pakize, Abolfazl Gandomi, "Comparative Study of Classification Algorithms Based on MapReduce Model", International Journal of Innovative Research in Advanced Engineering, ISSN: 2349-2163, Volume 1 Issue 7, August 2014, pg. no. 251-254.

[4] Mladen A. Vouk, "Cloud Computing – Issues, Research and Implementations", Journal of Computing and Information Technology - CIT 16, 2008, 4, 235–246 doi:10.2498/cit.1001391

[5] Hetal Bhavsar, Amit Ganatra, "A Comparative Study of Training Algorithms for Supervised Machine Learning", International Journal of Soft Computing and Engineering ISSN: 2231-2307, Volume-2, Issue-4, September 2012, pg. no. 74-81.

[6] Yugang Dai, Haosheng Sun, "The naive Bayes text classification algorithm based on rough set in the cloud platform", Journal of Chemical and Pharmaceutical Research, ISSN: 0975-7384, 2014, pg. no. 1636-1643.

[7] P Beaulah Soundarabai, Aravindh S, Thriveni J, K.R. Venugopal and L.M. Patnaik, "Big Data Analytics: An Approach using Hadoop Distributed File System, International Journal of Engineering and Innovative Technology, ISSN: 2277-3754, Volume 3, Issue 11, May 2014, pg. no. 239-244.

[8] Vijay D. Katkar, Siddhant Vijay Kulkarni, "A Novel Parallel implementation of Naive Bayesian classifier for Big Data", International Conference on Green Computing, Communication and Conservation of Energy, 978-1-4673-6126-2/2013 IEEE, pg. no. 847-852.

[9] Yogachandran Rahulamathavan, Raphael C.-W. Phan, Suresh Veluru, Kanapathippillai Cumanan, Muttukrishnan Rajarajan, "Privacy-Preserving Multi-Class Support Vector Machine for Outsourcing the Data Classification in Cloud", IEEE Transactions on Depedable and Secure Computing, Vol. 11, January 2014, pg. no. 1-14.