A Load Balancing Hierarchical Clustering Based Heart Disease Prediction Using Data Lake Architecture

Dilli Babu M.[1]
[1]Research Scholar, Department of Computer Science and Engineering,
Hindustan Institute of Technology and Science,
Chennai, Tamil Nadu 603103, India
deenshadilli@gmail.com

Sambath M.[2]
[2]Associate Professor, Department of Computer Science and Engineering,
Hindustan Institute of Technology and Science,
Chennai, Tamil Nadu 603103, India
msambath@hindustanuniv.ac.in

Article Info
Page Number: 870 – 882
Publication Issue:
Vol. 71 No. 3 (2022)

Abstract

The identification of illness is a vital and complex work in medicine. The identification of heartdiseasefrom varied features is a foremost concern which is not liberated from counterfeitbeliefalong with unpredictable effects. The healthcare industry congregates huge quantity of heart disease data that are stored in Data Lake and the gathered information utilized for effective diagnosing. Since the amount of stored data augment, mining the data becomes more significant in loading, locating, extracting, managing and querying information to offer improved medical care to patients, thus resulting in efficient diagnosis of the disease. The health care data is retrieved from varied data sources and it is usually large scale. The existing architecture like RDBMS related data management is futile. The Hadoop framework offers solutions related to enormous data processing. There exist cases of data missing during transit. The missing data can be replaced using imputation method. In this paper, the Expectation Maximization algorithm is used for preprocessing and non negative matrix factorization with hierarchical clustering (NMF- HC) algorithm is used for clustering. This paper also proposes a load balancing algorithm for heterogeneous MapReduce environment using the Hadoop simulator HSim. The results demonstrate an enormous enhancement in the performance of the simulated cluster. The proposed NMF-HC

Article History
Article Received: *12 January 2022*
Revised: *25 February 2022*
Accepted: *20 April 2022*
Publication: *09 June 2022*

algorithm produces promising result in predicting heart disease.
Keywords:- Data Lake, Hadoop framework, load balancing, clustering, data query.

1. INTRODUCTION

With the rapid progress of IT expertise, the Medical Informationization [1, 2] is getting more accepted by the end user. Conventional medical information platform based on Hospital Information System [3] gathers and controls the data of patients and hospitals.. In recent years, new invention of health care systems like EMR (Electronic Medical Record) [4] and PHR(Personal Healthcare Record) [5] store and administer thorough and complete health records of all patients. These data from EMR and PHR can be shared to health consultant and doctors. But with the increase in the users and advancement in platform, the data too increases exponentially. At the same time, storing, distributing, managing and sharing these enormous data turn out to be an research concern [6]. The health care data is retrieved from varied data sources and it is usually large scale. The existing architecture like RDBMS related data management is futile. The Hadoop framework offers solutions related to enormous data processing.

HDFS is based on the master slave architecture. It consists of a one NameNode and many DataNodes. The NameNode and DataNode are placed in racks. The NameNode acts as the master and the DataNode act as the slave. The NameNode manages and grants permission to files accessed by clients. The DataNode is a physical unit which stores data. The DataNode performs all file related operations.The files are stored in a distributed fashion in the HDFS. Files are partitioned into fixed sized blocks. Each block contains exactly three replicas. These replicas are stored in three diverse DataNodes. The replication of blocks augments the file access performance to a great level. If any node fails, then the replicated blocks grant fault-tolerance capacity. The DataNodes also take the responsibility as TaskTrackers and NameNode takes the responsibility as JobTrackers in the MapReduce framework. The JobTracker accepts all jobs for data processing and splits the task into small tasks that executes in the TaskTrackers. To handle the load imbalance issue, the in-built Balancer will be executed that moves the over utilized blocks of DataNodes to less utilized block of DataNodes. The movement of blocks is determined by the threshold set on that cluster. For example, when the HDFS Balancer tool runs at threshold 10% then 10% of utilization of the disk exists. In general, for a 60% cluster limit there exists an approximate disk utilization from 50 % to 70%.[7]. The problem with HDFS Balancer is it doesn't predetermine the threshold limit of block movement. Moving of data block incurs high data transmit capability such that the load balancing will be fast.

In the MapReduce model[8][9], the Hadoop framework processes huge data under the distributed environment. Applications based on distributed computing can be developed using the Hadoop framework. Three components are involved to achieve this-the HDFS, mappers and reducers. A number of intricate low-layer information such as hardware and software are required for the components to accomplish their task. The Hadoop framework provides a FIFO job scheduler which is homogenous based and varied clusters run on this environment. For

example, the Hadoop framework employed at Yahoo[10] consists of homogeneous clusters that has four thousand processors, Ram size of 3TB with storage size of 1.5PB. Based on these configurations, various benchmarks are set and competitions held to highlight the strength of the Hadoop computing environment. Nowadays clusters that are heterogeneous are emerging. The unbalanced load is an important factor in the heterogeneous clusters which does not exist in the homogenous environment. The cluster performance is mostly reliable based on the heterogeneities of the environment.

A distributed algorithm [11] categorizes the DataNodes of HDFS into a Chord ring. Using this ring arrangement, the DataNodes balances itself without the interference of NameNode and the load reorganization need not be done by the DataNodes which increases the performance of the system. In the heterogeneous clusters, a dynamic load balancing algorithm can be implemented. The algorithm employs sliding window method to accomplish significant enhancement of the Hadoop job processing [12]. There exists many Hadoop tools for load balancing. In [13], a custom block placement policy Hadoop tool is employed for load balancing by taking into consideration hardware constraints of the nodes.

Presently, a small amount of researches based on load balancing exist for the Map Reduce environment. The job execution time can be limited by the mapper due to some issues involved in reading and writing of local hard disk, overheads incurred while copying data [14]. Google Distributed File System is used to study the effect of un-balanced load problem [15]. The mapper and the reducer highly influence the Hadoop framework performance [16]. Genetic algorithm can be used to resolve the load balancing problem. In [17] , Parallel Hybrid PSO-GA based on genetic algorithm is used for the performance optimization of the Hadoop framework.

Medical blunders remain equally expensive and destructive [18]. Errors in medical data lead to more deaths than the death caused due to accidents and highly infectious diseases [19]. Research study highlights that on an average 195000 people out of 37 million patient records die in the United states due to errors in medical hospital records that are preventable [20]. Patients with Cardiovascular disease are prone to death faster than any other disease [21]. Hence there is an urge in the medical field to detect and prevent the cardiovascular diseases. There is an urgent need in the medical field to reduce the diagnosis time and at the same time to increase the prediction of diseases correctly, particularly in the detection and prediction of heart diseases [22]. Decision support system was introduced in the early days for this purpose that were based on statistical theory [23]. In recent days, more accurate prediction with zero error diagnosis is expected by patients. The health care industry is striving to give the best service in terms of quick and perfect prediction of diseases to patients. The best medical service industry is looked by the patients to undergo their treatment. The health care industry is providing quality service to patients in terms of best treatment to patients [24]. Hospitals are switching to computer based decision support system to reduce the cost incurred through clinical test. Majority of hospitals manage their service through computerized hospital information system for maintaining their patients record [25].

2. METHODOLOGY

The data lake is provided with data from varied data source. The varied data from health providers, health disease dataset, and patient generated data are stored in the data lake.

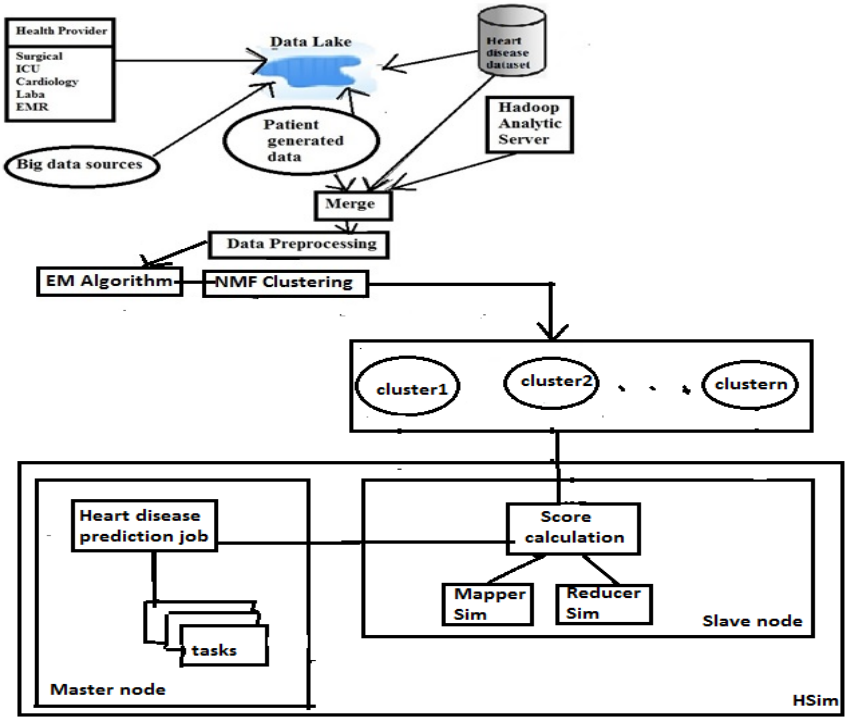

Figure 1: Workflow of the proposed system

The Expectation Maximization algorithm is used for preprocessing and NMF- hierarchical clustering (HC) algorithm is used for clustering. The different clusters are given as input to the Hadoop framework. The Hadoop framework performs better in terms of data uploading, massive data query and data management. The workflow of the proposed system is given in Figure 1.

The proposed work focuses on imputation method for the missing attribute in the dataset. Here the 13 attributes from the Cleveland database are considered. The missing attributes are reinstated with probable values based on Multi-Cycle Expectation Conditional Maximization (MCECM). The MCECM is based on matrix variant normal where the rows and columns have a different meanvector and covariance matrix. The next step is the clustering process. A new rank 2 NMF is used for the clustering process. For the computation of rank 2 NMF, a non negative least square method is applied. The output from this step determines where the leaf node should be processed or not and which of the leaf node needs to be processed.

2.1 NMF Hierarchical Clustering

The rank 2 NMF is used to split the preprocessed samples present in Cleveland dataset from the missing attribute imputation technique. The finding of the next node for imputation is the

challenging task. The strength of the clustering process is based on the partitioning of the dataset. The preprocessed data undergoes a splitting process. A score is computed at each leaf node. The leaf node is evaluated to check for the presence of two different clusters. For this purpose, the rank 2 NMF is used to determine which one of the clustering process to split at which leaf node. The rank 2 NMF is used recursively on the data that is preprocessed to generate the hierarchical tree structure. The proposed NMF hierarchical clustering algorithm introduces the split step for choosing the leaf node based on the cluster label. The NMF hierarchical clustering uses the rank 2 NMF to compute the score of leaf nodes. The rank 2 NMF is rum two columns of C The current leaf node is then selected that has the large score and this leaf node is taken for the next level of split. Initially a leaf node N is split only if it contains two different classes that can be split from that node. This leaf node N gains a high score on the positive class. In general, if the leaf node N gains a high positive score among the preprocessed dataset samples then it will be considered for further split by the algorithm. The leaf node with the high positive class score is split as left else it is split as right in the hierarchical tree. The normalized discounted cumulative gain (NDCG) is used for this purpose which takes the value between 0 and 1.

For an attribute in column 'c', a leaf node L is related with a distribution of c_N by running the rank 2 NMF algorithm generating the node L. Then a list of all positive and negative class is obtained. The list is then ranked in descending order depicted by des_N. In the same way, the left children of the node denoted by L_{left} and right children of the node denoted by R_{right}rank list is also obtained. The $child_L$and $child_R$ is the rank list of children lest and children right respectively. A modified normalized discounted cumulative gain(mNDCG) score is proposed based on $des_{L\ and}$ des_R assuming $child_N$ is in ideal ranked list.

Let the data points in the clustering $child_N$be $dt_{l1},\ldots,dt_{ln}$ $\ldots dt_{r1},\ldots dt_{rn}$.Then for each cluster ponts, position discount factor and the gain factor is given by equation (1) and equation (2).

$$p(dt_j) = \log(n-\max\{j_1,j_2\}+1) \quad (1)$$

$$g(dt_j) = \frac{\log(n-j+1)}{p(dt_j)} \quad (2)$$

Where $lj_1=rj_1=j$. For every data point dt_j, the positions j_1,j_2 is found among the unordered list. The shuffled orderings on both left and right side is given in equation (3). Eq (3) represents the gain of the data points dt_jthat are preprocessed. The orderings in descending order is given in equation (4).

$$\{g(dt_j)\}_{j=1}^{m} \quad (3)$$

$$\{\hat{g}(dt_j)\}_{j=1}^{m} \quad (4)$$

Then the shuffled ordering dt_s ($dts=dt_L$ or dt_R), the mNDCG is given by equation (5)

$$mDCG(dt_s)=g(dt_{n1}) + \sum_{j=2}^{m} \frac{g(dt_{n_i})}{log_2(j)} \quad (5)$$

$$mIDCG=\hat{g} + \sum_{j=2}^{m} \hat{g}/log_2(j) \quad (6)$$

$$mNDCG(dts)=\frac{mDCG(dt_s)}{mIDCG} \quad (7)$$

The leaf node's score is given in equation (8)

$$Score(N) = mNDCG(dt_L)\ X\ mNDCG(dt_R) \quad (8)$$

Some useful findings based on the score calculated above can be done. Assume that the children Left and Right depict different diseases, then a chosen attribute will be in high score in one among the following: dt_L and dt_R. Hence the peak word does not belong to the heavy discount. In this case, $mNDCG(dt_L)$ and $mNDCG(dt_R)$ can have more huge values. In the case of children Left and Right depicting same diseases, the peak words belong to heavy discount and $mNDCG(dt_L)$ and $mNDCG(dt_R)$ can have very less values. In the case where the children Left illustrate the same disease based on the dataset samples from the leaf node that are preprocessed and the children Right illustrate a completely dissimilar class then $mNDCG(dt_L)$ can have large values with $mNDCG(dt_R)$ having less values.

2.2 Load Balancing

The optimization problem can be solved using Genetic algorithm where the solutions can be characterized in terms of chromosomes represented as binary strings. But the binary depiction is not practicable when the number of mappers in a Hadoop cluster setting is on the whole huge which lead to the input of very long binary strings. In Hadoop environment, the total time (T_{map}) of a *mapper* in handling a data portion is given in Eqaution (9).

$$T_{map} = d_c + d_p + d_m + d_b \quad (9)$$

d_c is data copying time, the time involved in copying a data chunk from HDFS to local hard disk. d_p is the running time involved in processing a data by the processor. d_m is the merging time involved in merging the mapper output files into a single file. d_b is the buffer spilling time in clearing the filled buffers.

$$d_c = \frac{S_d}{min(W_h, N_b)} \quad (10)$$

Where S_d is data chunk size, W_h is hard disk writing speed in MB/second, N_b is the network bandwidth in MB/second.

$$d_p = \frac{S_d}{R_p} \tag{11}$$

R_p is the running processor speed in MB/second

$$t_m = \frac{D_m \times R_a \times N_f}{H_d} \tag{12}$$

$$d_m = \frac{S_d X R_s X F_n}{W_h} \tag{13}$$

R_s is the ratio of intermediate data size to data chunk size. F_n is the number of frequencies involved for intermediate data processing. B_n is the number of times the buffer gets filled. B_s is the size of the mapper buffer.

$$d_b = \frac{B_n X B_s}{W_h} \tag{14}$$

The total time T_t in processing the data chunks using a MapReduce Hadoop is the maximum of the time involved by including all the 'j' mappers that participate in the processing.

$$T_t = \max (T_1, T_2, T_3, \ldots, T_j) \tag{15}$$

Based on divisible load theory[15][17], all the data processing by the mappers need to be completed at the same time to achieve minimum T_t.

If T_i is the processing time of i^{th} mapper , then the average time $\bar{T}$ required for the 'j' mappers for data processing is given by Equation (16).

$$\bar{T} = \frac{\sum_{i=1}^{j} T_i}{j} \tag{16}$$

The fitness function is given in Equation (17)

$$f(T) = \sqrt{\sum_{i=1}^{j} (\bar{T} - T_i)^2} \tag{17}$$

The heterogeneity of the cluster is given in Equation (18)

$$\text{Heterogeneity} = \sqrt{\sum_{i=1}^{j} (\bar{c} - c_i)^2} \quad (18)$$

Where 'c' is the total processing speed of the cluster , $\bar{c}$ is the average processing speed of the cluster, c_i is the processing speed of the i^{th} processor and 'j' is the number of processor involved in the cluster.

3 EXPERIMENTAL RESULT

The dataset is taken from the UCI repository for our experimentation. The different attributes in the dataset are Age, sex, chest pain type, resting bloodpressure, serum cholesterol in mg/dl, fastingblood sugar, resting electrocardiographic results,and maximum heart rate achieved, exerciseinduced angina, ST depression, and slope of thepeak exercise ST segment, number of majorvessels. In the proposed system, 909 records are taken in which 455 records are used as training dataset and 454 records used as testing dataset. The target attribute is the 'diagnosis' attribute. The diagnosis attribute is set to '1' if the patient is diagnosed with heart disease and the diagnosis attribute is set to '1' if the patient is not diagnosed with heart disease. Table 1 depicts the attributes of Cleveland Heart Disease database.

Clustering Accuracy: It is used to determine the results obtained in clustering. The clustering accuracy (q) is given in equation (19).

$$q = \frac{\sum_{j=1}^{k} a_j}{n} \quad (19)$$

k is the number of clusters,a_i is the number of instances. Error of the cluster is e=1-q

Table 1: Attributes of Cleveland Heart Disease database.

Attributes	Description	Type
Age	age in years	Numerical
sex	sex (1 = male; 0 =female)	Categorical
cp	chest pain type Value 1:typical angina Value 2:atypical angina Value 3:nonanginal pain Value 4:asymptomatic	Categorical
restbps	resting blood pressure (in mm Hg on admission to the hospital)	Numerical
chol	serum cholestoral in mg/dl #10 (trestbps)	Numerical
fbs	(fasting blood sugar > 120 mg/dl) (1 = true; 0 = false)	Categorical
restecg	Resting electrocardiographic results	Categorical
thalach	maximum heart rate achieved	Numerical

exang	exercise induced angina (1 = yes; 0 = no)	Categorical
Oldpeak	ST depression induced by exercise relative to rest	Numerical
slope	the slope of the peak exercise ST segment Value 1: upsloping, Value 2: flat and Value 3:downsloping	Categorical
ca	number of major vessels (0-3) colored by flourosopy	Categorical
thal	3 = normal; 6 =fixed defect; 7 =reversable defect	Categorical
num	diagnosis of heart disease . Value 1: present Value 0: not_present	Categorical

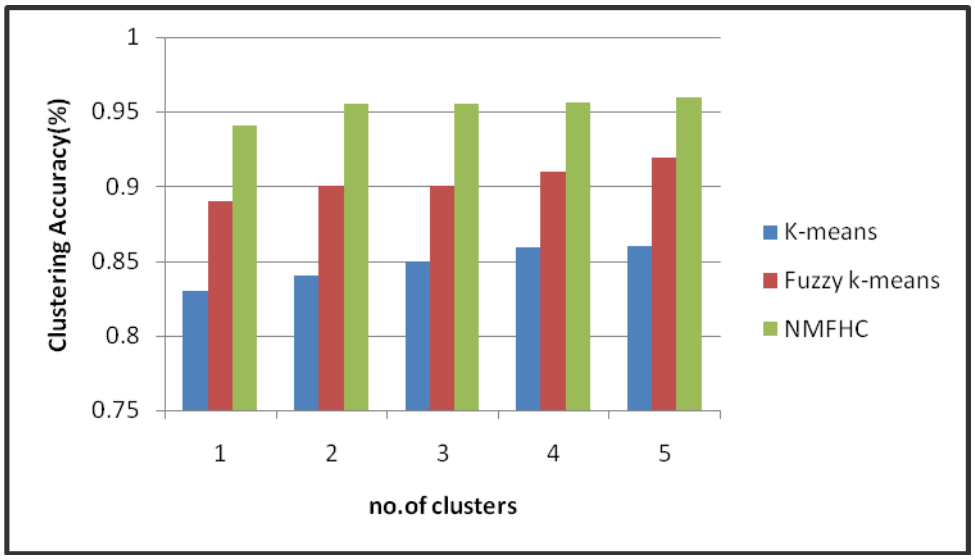

Figure 2: Clustering accuracy

Figure 2 depicts the clustering accuracy of the existing and the proposed NMFHC methods. The clustering accuracy is more for the proposed method than the existing methods. Since the proposed method contains the preprocessing step of replacing values for missed attribute data values, the clustering accuracy gives promising results. Figure 3 gives the error rate of the existing and proposed NMFHC method. It can be inferred from figure 3 that the error rate is less for the proposed system than the existing methods.

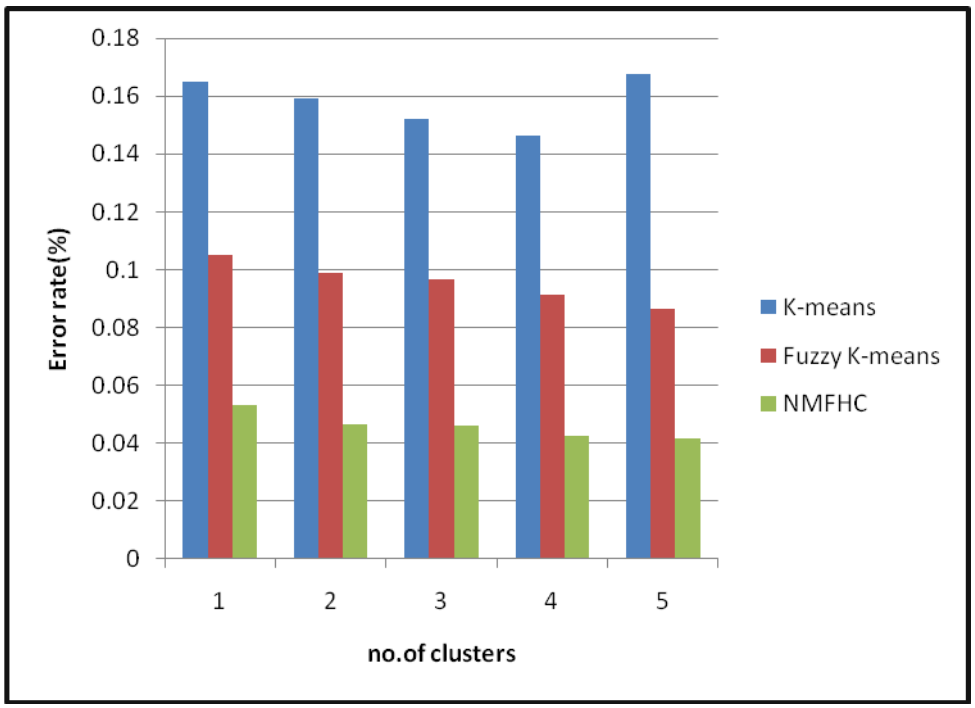

Figure 3: Error rate

In our experimental results for load balancing, the simulated algorithm MR-LSI [16] is used. The number of nodes for simulation is taken as 20. In each node, one processor is available for data processing. There exists two cores in each of the processors. The size of the data Test1 was 10GB and that of Test2 ranged from 10GB to 100 GB. The number of heterogeneities ranged from 0 to 2.48. There was one hard disk in each node. The hard disk had a reading speed of 80MB/s and writing speed of 40 MB/s. The number of Map instances was 2 and reduce instances was 1. The sort factor was fixed at 100. Initially 10GB of data was tested in the simulated cluster against different levels of heterogeneity. The level of heterogeneity in Figure 6 is lower than 1.18 that does not highlight any heterogeneity which in turn does not show improvement in the MR-LSI performance based on load balancing. But the application of Load balancing gives a reduction in the MR-LSI overhead linearly with increase in the heterogeneity levels.

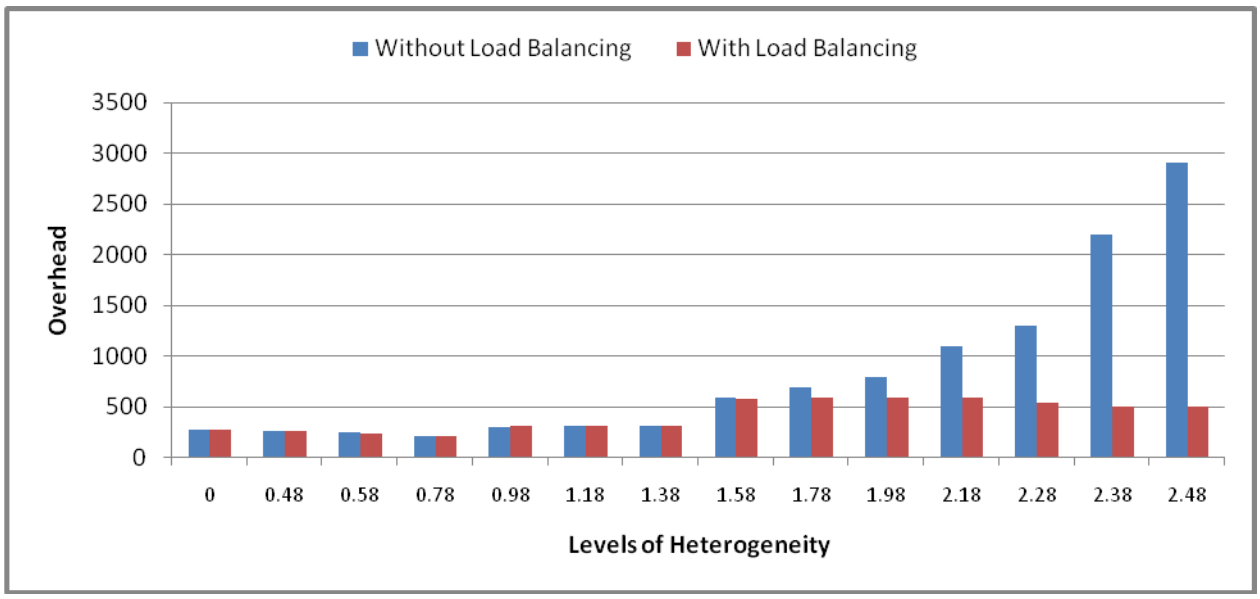

Figure 4: Levels of heterogeneity

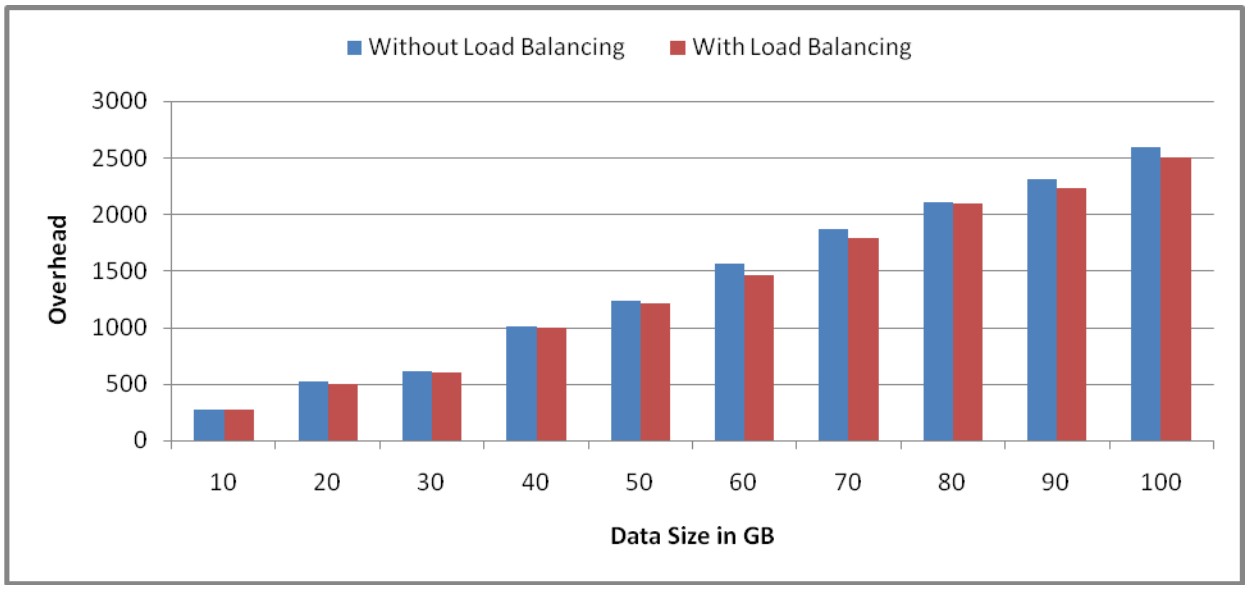

Figure 5: Data sixe

The above figure (5) depicts the overhead incurred with the increase in data size. As the size of the data increases, the overhead also increases. But when load balancing is applied, the overhead is somewhat better than without load balancing.

CONCLUSION

In this paper, the Expectation Maximization algorithm is used for preprocessing and non negative matrix factorization with hierarchical clustering (NMF- HC) algorithm is used for

clustering. This paper also proposed a load balancing algorithm for heterogeneous MapReduce environment using the Hadoop simulator HSim. The results demonstrate an enormous enhancement in the performance of the simulated cluster. The proposed NMF-HC algorithm produced promising result in predicting heart disease.

References

[1] Wei Liu, and Dedao Gu, "Research on construction of smart medical system based on the social security card", 2011 International Conference on Electronic and Mechanical Engineering and Information Technology (EMEIT), pp. 4697–4700, 2011.

[2] Xiaolin Lu, "Design and implementation of cooperative distributed dental medical information system", Proceedings of the Ninth International Conference on Computer Supported Cooperative Work in Design, vol 2, pp. 799–803, 2005.

[3] Boqiang Liu, Xiaomei Li, Zhongguo Liu, et al. "Design and implementation of information exchange between HIS and PACS based on HL7 standard", International Conference on Information Technology and Applications in Biomedicine, pp.552–555, 2008.

[4] Yuwen Shulim, Yang Xiaoping, Li Huiling, "Research on the EMR Storage Model", International Forum on Computer Science-Technology and Applications, pp.222–226, 2009.

[5] Faro, A., Giordano, D., Kavasidis, I., Spampinato, C., "A web 2.0 telemedicine system integrating TV-centric services and Personal Health Records", 10th IEEE International Conference on Information Technology and Applications in Biomedicine (ITAB), pp.1–4, 2010.

[6] David Kaelber and Eric C Pan, "The Value of Personal Health Record (PHR) Systems", AMIA Annu Symp Proc., pp.343–347, 2008.

[7] "Hadoop Fair Scheduler". https://hadoop.apache.org/docs/r2.7.2/hadoopyarn/had oop-yarn-site/FairScheduler.html. Accessed 20 May 2017.

[8] Dean, J., and Ghemawat, S. (2004). MapReduce: Simplified Data Processing on Large Clusters. In Proc. of OSDI'04: Sixth Symposium on Operating System Design and Implementation, San Francisco, CA.

[9] Lämmel, R. (2007). Google's MapReduce programming model —Revisited. Sci. Comput. Program. vol. 68, pp 208-237.

[10] Yahoo. Hadoop at Yahoo! Available at:http://developer.yahoo.com/hadoop/. (Lasted accessed: 20-Nov-2010).

[11]Hsiao, H-C., Chung, H-Y., Shen, H. and Chao, Y-C. (2013) 'Load rebalancing for distributed file systems in clouds', *IEEE Transactions on Parallel and Distributed Systems*, Vol. 24,pp.951–962.

[12] Liu.Q, Cai.W, Shen.J, Fu.Z, Liu.X and Linge.N (2016) "A speculative approach to spatial-temporal efficiency with multi-objective optimization in a heterogeneous cloud environment", *Security and Communication Networks*, vol. 9, no. 17, pp. 4002- 4012.

[13] Liu.Y, Jing.W,Liu.Y, Lv.L,Qi.M and Xiang.Y(2016) "A sliding window-based dynamic load balancing for heterogeneous Hadoop clusters", *Concurrency and Computation: Practice and Experience*, vol. 29, no. 3, p. e3763.

[14] Groot, S. (2010). Jumbo: Beyond MapReduce for Workload Balancing. VLDB 2010 , 36th International Conference on Very Large Data BasesSingapore.
[15] Ghemawat, S., Gobioff, H., and Leung, S.T. (2003). The google file system. In SOSP '03, pp 29-43, New York, NY, USA.
[16] Li, H., Wang, Y., Zhang, D., Zhang, M., and Chang, E. Y. (2008). Pfp:parallel fp-growth for query recommendation. In RecSys '08, pp 107-114,New York, NY, USA.
[17] Sadasivam, G. S., and Selvaraj, D. (2010). A Novel Parallel Hybrid PSO-GA using MapReduce to Schedule Jobs in Hadoop Data Grids.2010 Second World Congress on Nature and Biologically Inspired Computing, Kitakyushu, Fukuoka, Japan.
[18] Hall, J., First, make no mistakes. The New York Times. (2009).
[19] SoRelle, R., Reducing the rate of medical errors in the United States. (2000).
[20] Patient Safety in American Hospitals Study Survey by HealthGrades, 2004.
[21] CDC‟s report, http://www.cdc.gov /nccdphp/ overview. html.
[22] Yan, H.-M., Jiang, Y.-T., Zheng, J., Peng, C.-L., & Li, Q.-H. A multilayer perceptron-based medical decision support system for heart disease diagnosis. (2006)
[23] http:/astrosun.tn.cornell.edu/staff/loredo/bayes/
[24] Herbert Diamond, Michael P. Johnson, Rema Padman, Kai Zheng, “Clinical Reminder System: A Relational Database Application for Evidence-Based Medicine Practice ”INFORMS Spring National Conference, Salt Lake City, 2004.
[25] Sellappan Palaniappan , Rafiah Awang “Web-Based Heart Disease Decision Support System using Data Mining Classification Modeling Techniques” Proceedings of WAS2007