Data Mining Using Hierarchical Agglomerative Clustering Algorithm in Distributed Cloud Computing Environment

Kriti Srivastava, R. Shah, D. Valia, and H. Swaminarayan

***Abstract*—Today increase in worldwide business led to offices distributed across geographical location .Hence data are loosely distributed across regionalized large scale databases across regionalized offices. To perform data mining it is required to merge distributed data and perform data mining algorithm on it. Cloud computing poses a diversity of challenges in data mining operation arising out of the dynamic structure of data distribution as against the use of typical database scenarios in conventional architecture. This document presents a way to implement Hierarchical Agglomerative Clustering Algorithm in such way so as to make it suitable for large dataset and increase its efficiency by executing task in parallel. The result shows that with increase in data set linear growth of execution time.**

***Index Terms*—Star cluster, hierarchal agglomerative clustering, virtual k mean, cloud computing.**

I. Introduction

Increase in the usage of cloud computing has sparked a new interest among researchers of data mining. Using contemporary algorithms has proven to be inefficient on the cloud. It is not suited for large and highly distributed database because the time for execution is very large. [1] Cloud has emerged as a computing infrastructure that enables rapid delivery of computing resources as a utility in a dynamically scalable virtualized manner [2]. Data mining is a process of discovering meaningful patterns and relationships that are hidden in large data set [3]. Simply stated data mining refers to extracting or “mining” knowledge from large amounts of data [4]. Clustering is the process of grouping a set of physical or abstract objects into classes of similar objects. A cluster is collection of data objects that are similar to one another within the same cluster and are dissimilar to the objects in other clusters. Though there are many algorithms which takes care of large database but huge memory usage is always a concern. Using cloud to process and store database can solve this problem as it can take care of more memory requirement very easily.

A hierarchical clustering method works by grouping data objects into a tree of clusters. Hierarchical clustering methods can be further classified as either agglomerative or divisive, depending on whether the hierarchical decomposition is formed in a bottom-up (merging) or top-down (splitting) fashion [5].

In this paper we will focus on Hierarchical Agglomerative bottom up merging fashion based algorithm and suit it to geographical distributed data set. Our aim is to increase the efficiency of agglomerative clustering algorithm as well as to make it suit for large data. To implement this we require the cloud computing virtualized environment [6]. Virtualization is a key technology used in data centers to optimize resource [7]. Assume data distributed among different node. By virtualization we create instances of each geographical distributed node. This bottom-up strategy starts by placing each object in its own cluster and then merges these atomic clusters into larger and larger clusters, until all of the objects are in a single cluster or until certain termination conditions are satisfied.

The paper consists of five sections: Section I provides introduction on cloud computing, Hierarchical Agglomerative Clustering and Virtualization concept. Section II describes the design of modified agglomerative clustering technique and algorithm that suit for cloud platform. Section III describes the experimental setup to implement on cloud based architecture. Section IV provides us with experimental results and benefits on implementing it. Section V describes the conclusion and future work to be performed.

II. Design of Efficient Agglomerative Clustering Techniques

It has been argued that to perform effectively on large databases, the algorithm should require no more than one scan of the database, have the ability to provide “best “ answer so far, be suspendable, stoppable and resumable, be able to update the results incrementally etc. Keeping these points in mind the basic idea would be to read the subsets of database, apply clustering algorithm and combine the results with those from prior samples and proceed in this way till all the data is available in main cluster.

The hierarchical clustering algorithm is suited for small dataset but for making it suite to large dataset. [8] We will divide it in two tasks - 1. Microclutering stage 2. Macroclustering stage. As shown in Fig. 1. Modified Hierarchical Agglomerative clustering perform processing at three layers.

A. Apply Virtual K Mean

Layer 1: In this layer data from various geographical distributed dataset are loaded into individual virtualized node. Then we apply virtual k-mean algorithm on each node which will form k number of cluster on individual node. This output will be stored on separate file created at individual node. Thus macroclustering occurs at this layer.

B. Merging Files

Layer 2: In this layer the outputted files which consist of

Manuscript received August 9, 2012; revised December 16, 2012.

Kriti Srivastava is with the D. J. Sanghvi College of Engineering, Mumbai, India (e-mail: kriti.srivastava@djsce.ac.in).

R Shah is with the Capgemini, Mumbai, India (e-mail: ronshah123@gmail.com).

D. Valia is with the Sokrati, Pune, India (e-mail: dwalia@gmail.com).

DOI: 10.7763/IJCTE.2013.V5.741

k-centriod and cluster are merging into single file called Master file. To reduce any error normalization is performed on this master file. Thus master file contain data which are cluster analysis and outlier error free.

C. Hierarchal Agglomerative Clustering (HAC)

Layer 3: In this layer on the outputted master file we will apply Hierarchical agglomerative clustering algorithm. Thus the output will be in form of dendogram. Thus microclustering occurs at this layer.

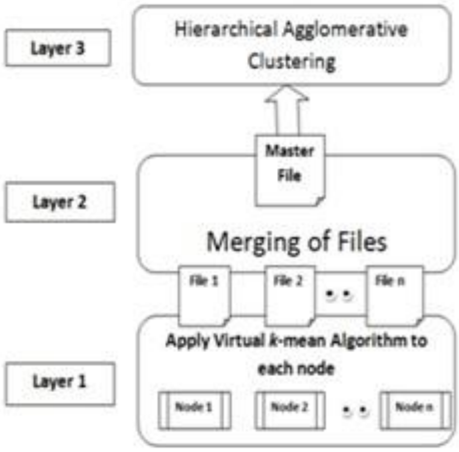


Fig. 1. Design

D. Algorithm

Step 1: Define the number of nodes N which would be equal to geographical distributed dataset.

Step 2: Apply k-mean clustering algorithm on each node individually.

Step 3: The output will be each file consists of k number of centroid and respective cluster stored in separate files.

Step 4: Perform merging of separated files in to single master file. Thus single master file consist of (k * n) of centroid.

Step 5: Perform normalization on master file to reduce error by outlier and cluster analysis.

Step 6: Apply Hierarchical Agglomerative Clustering algorithm on master file which will output dendogram as shown in below Fig. 2.

Thus in Fig. 2 A, B, C, D and E represent centroid and data cluster represent the cluster formed by grouping data objects using k mean algorithm.

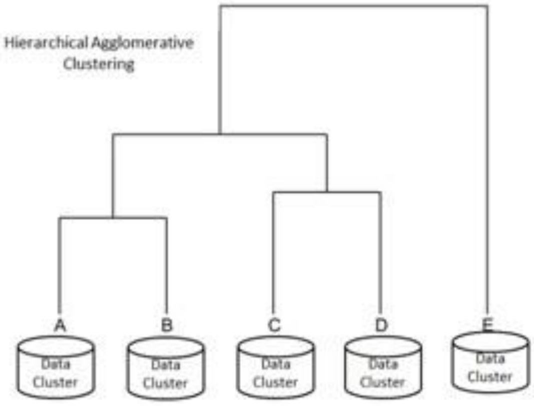


Fig. 2. Dendogram

The main difference between the normal algorithm and modified is as shown in output figure 2. The HAC has A, B, C, D and E represent individual cluster while modified algorithm represents central centroid of data cluster generated by k mean algorithm. Thus this modification provides us with following benefits:

1) We will be able to use Hierarchical Agglomerative clustering algorithm for large set of data.[9]
2) The efficiency of the algorithm has been increase due to performing macroclustering on large data set followed by microclustering on outputed centroid of data cluster.
3) Parallelism of task reduce the time required for execution.

III. Experimental Setup

The implementation of the above concept in cloud architecture requires master and slave node. [10] For master and slave architecture we have used MIT's StarCluster platform whose logical structure is shown in Fig. 3 [6]. StarCluster creates Amazon EC2 instance for master and all the nodes. It enables SSH access between them [11] the memory blocks between them are shared by NFS.

The nodes have MySQL and Java pre-installed. The data sets have been stored in MySQL and the modified algorithm has been written in JAVA.

On the master node, we execute commands on the nodes by using SSH to ensure that the commands are run in a parallel manner. We pass the commands to the qsub of Sun Grid Engine.

The k-mean algorithm gets executed on each slave node. Thus the task required to be executed by layer 1 shown in Fig. 1 gets executed at slave node.

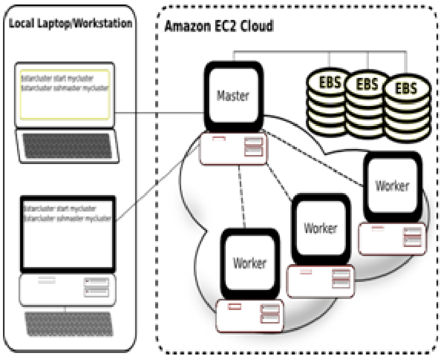
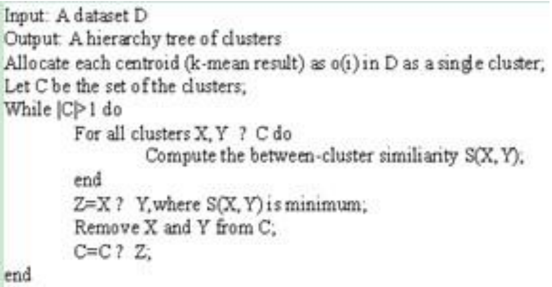


Fig. 3. Logical structure

The result of virtual k-mean from each node is transferred to the Master Node. After normalizing the result HAC algorithm is run on the master to obtain the final results. The sample algorithm for executing Hierarchical Agglomerative clustering is shown in Fig. 4. Thus task required to be executed by layer 2 and layer 3 shown in Fig. 1 gets executed at master node.



```
Input: A dataset D
Output: A hierarchy tree of clusters
Allocate each centroid (k-mean result) as o(i) in D as a single cluster;
Let C be the set of the clusters;
While |C|>1 do
        For all clusters X, Y ? C do
                Compute the between-cluster similiarity S(X, Y);
        end
        Z=X ? Y, where S(X, Y) is minimum;
        Remove X and Y from C;
        C=C ? Z;
end
```

Fig. 4. Sample code for HAC algorithm

IV. Result Analysis

We have applied the modified HAC algorithm on sample medicine database having attributes weight and ph [12]. To compare its efficiency we have executed with variation of nodes. The total number of data mined are number of nodes*number of rows in each node. Thus as nodes increase the data also increases but doesn't deteriorate the performance as shown in Fig. 5. There is a linear increase in time required for execution. With quadratic increase in data across cloud environment the time required for execution increases linearly. Hence efficiency of the modified algorithm has been increase greatly by parallelism of task on cloud based architecture.

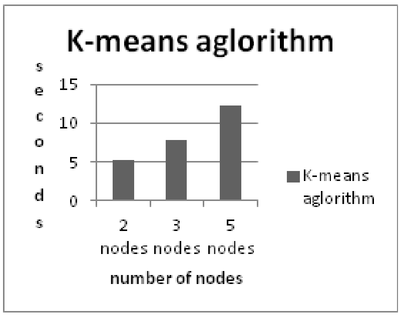


Fig. 5. Result analysis.

V. Conclusion and Future Work

Cloud is a highly dispersed computing model which is having inherent advantages of scalability, availability, elasticity, pay per use etc. Thus modifying algorithm to suit cloud architecture can enables above benefits. In addition to this it provides us with following benefits like Hierarchical Agglomerative clustering can handle large dataset, increase efficiency of algorithm and parallelism has reduce time required for execution. Thus the result shows that cloud architecture is providing additional advantages for data mining. In future we can compare the results obtained from cloud platform with mapreduce framework to understand the effectiveness.

References

[1] Z. X. Hou, X. S. Zhou, J. H. Gu, Y. L. Wang, and T. H. Zhao, "ASAAS: Application software as a service for high performance cloud computing," in *Proc. of 2010 12th IEEE International Conference on High Performance Computing and Communications (HPCC)*, pp. 156-163, 2010.

[2] W. T. Tsai, X. Sun, and J. B. Sooriya, "Service oriented cloud computing architecture," in *Proc. IEEE 2010 Seventh International Conference on Information Technology.*

[3] U. Fayyad, G. P. Shapiro, and P. Symth, " From data mining to knowledge discovery in databases," 0738-4602-19196, A Magzine37-53

[4] Data Mining and Analytics Resources. [Online]. Available: http://www.kdnuggets.com/gpspubs/aimagkdd-overview-1996-Fayyad.pdf

[5] J. W. Han and M. Kamber. *Data Mining Concepts and Techniques*. Morgan Kaufmann Publishers, Champaign CS497JH, fall 2001. [Online]. Available: http://www.cs.uiuc.edu/~hanj/bk2/

[6] Logistic Regression and Newton's Method. [Online]. Available: http://www.stat.cmu.edu/~cshalizi/350/ lectures/08/lecture-08.pdf

[7] T. R. G. Nair and K. L. Madhuri, "Data mining using K-mean integrating data fragment in cloud environment," *IEEE.*

[8] S. Pippal, V. Sharma, S. Mishra, and D. S. Kushwaha, "Secure and efficient multitenant database for an ad hoc cloud," *Securing Services on the Cloud (IWSSC)*, pp. 46-50, 2011.

[9] J. Z. Wang, J. G. Wan, Z. Liu, and P. Wang, "Data mining of mass storage based on cloud computing," in *Proc. of 2010 9th International Conference, Grid and Cooperative Computing (GCC)*, pp. 426-431, 2010.

[10] M. Comerio, H.-L. Truong, and C. Batini, "Dustdar, S, cloud service engineering; Service-oriented computing and applications (SOCA)," *2010 IEEE International Conference on Digital Object Identifier*, pp.1-6, 2010.

[11] K. R. Jackson, L. Ramakrishnan, K. Muriki, S. Canon, S. Cholia, J. Shalf, H. J. Wasserman, and N. J. Wright, "Performance analysis of high performance computing applications on the amazon web services cloud," in *Proc. IEEE Second International Conference on Cloud computing technology and science (Cloud Com)*, pp.159-168, 2010.

[12] Performance versus Cost of a Parallel Conjugate Gradient Method. [Online]. Available: http://web.mit.edu/star/cluster/docs/latest/overview.html

Kriti Srivastava is an assistant professor in D. J. Sanghvi College of Engineering, Mumbai, India. She has completed her Masters in Computer Science from NMIMS University, Mumbai. Her research area and study interest are distributed computing, data mining, business Intelligence and Cloud Computing.

R. Shah is from Capgemini, Mumbai, India. He had Completed his Bachelors in Information Technology from Mumbai University in July 2012. His research area and study interest are Networking, Distributed computing and Cloud Computing.

D. Valia is from Sokrati, Pune, India. He had Completed his Bachelors in Information Technology from Mumbai University in July 2012. His research area and study interest are Distributed computing, data mining and cloud computing.

H. Swaminarayan is a fellow at Teach for Indiai. B.E in Information Technology from D. J. Sanghvi College of Engineering.