ISSN (Print): 0974-6846 ISSN (Online): 0974-5645

Exploring Non-Homogeneity and Dynamicity of High Scale Cloud through Hive and Pig

Kashish Ara Shakil*, Mansaf Alam and Shuchi Sethi*

Department of Computer Science, Jamia Millia Islamia, New Delhi - 110025, India; shakilkashish@yahoo.com, malam2@jmi.ac.in, shuchi.sethi@yahoo.com

Abstract

Cloud environment is usually associated with non-homogeneity and dynamicity in terms of resource usage and access at all levels. The study of this heterogeneous and non-uniform behavior is therefore an important problem. Google cluster trace which is a production trace released by Google in November 2014 serves as an example of a high scale Cloud environment. This paper deals with statistical analysis of this cluster trace. Since the size of production trace is very huge therefore, Hive which is a HadoopDistributed File System (HDFS) based platform for querying and analysis of big data, has been used. Hive was accessed through its Beeswax interface. The data was imported into HDFS through HCatalog. Apart from Hive, Pig which is a scripting language and provides abstraction on top of Hadoop was used. The method adopted deals with clustering and studying the distribution of arrival time of jobs, distribution of resource usage and also study of distribution of process runtime. To the best of our knowledge the analytical method adopted by us is novel. The findings revealed that jobs in a production trace can be classified into major, mediocre and minor resource usage types. Furthermore, it can be concluded from our study that arrival time of jobs followed weibull distribution. Usage of resources such as CPU and memory was observed to be following a zipf like distribution while study of process runtime shows that some jobs had very small values of runtime while others had very large values hence they followed heavy tailed distribution. Our analysis will help researchers in properly understanding the nonhomogenous and dynamic behavior synonymous with cloud environment. It will also help them in developing new algorithms for resource allocation and scheduling in Cloud.

Keywords: Dynamicity, Hadoop, High Scale Cloud, Hive, Pig, Non-Homogenous

1. Introduction

Cloud computing is a technology that provides computing infrastructure¹, software and platform as a service to its end users^{2,3}. Cloud computing is at the moment becoming synonymous with non-homogeneity and dynamicity. There occurs non-homogeneity in type of resources available as well as their usage, leading to pitiable system performance and underutilized resources therefore it's important to characterize and study the workloads in cloud so as to properly allocate resources and achieve good performances in such a highly heterogeneous environment.

The major contributions of this paper are:

• Clustering of Jobs based on Resource Utilization.

- Clustering of Arrival Time of Jobs.
- Analysis of distribution of arrival time of jobs.
- Analysis of distribution of jobs based on resource usage.
- Analysis of distribution of process runtimes.

The work carried out in this paper will assist in further research that will be carried out in a heterogeneous and dynamic environment such as cloud. It will help the researchers in simulating cloud workloads and also in predicting the behavior of applications in cloud. It will also assist in optimizing allocation of resources and management of data in a cloud like environment. Nevertheless work has been carried out for management of data in cloud through cloud database management system architecture⁴ andK-median clustering⁵ but study

^{*} Author for correspondence

of a huge workload such as that of Google will further assist in this quest.

Many attempts have been made in order to understand the non-homogeneity and dynamicity of cloud environment. Most of these studies are based on analysis of a large workload such as that of Google. Google cluster is a workload which is a representative of cloud environment. It is highly dynamic and heterogeneous in nature. Several interesting observations can be made by analysis of such large production clusters which can aid promote in making scheduling decisions and improving the overall performance of the cluster as a whole.

Apart from study and analysis of Google cluster workload, in^{6,7}analysis of a Map Reduce production cluster has been done. They have analyzed Yahoos Trace data collected from Yahoos M45 supercomputing cluster and has logs of about 10 days. This cluster has around 400 nodes, 4000 processors and approximately 3 terabytes of memory. They have identified resource utilization patterns, sources of failures of job and job patterns. According to them jobs in cluster followed a long tailed distribution and they have also observed the behavior of users that users run the same job repeatedly and there also exists large error latency in jobs in this cluster.

In⁸ analysis of Google trace has been performed, and the authors have concluded that there occurs heterogeneity in all the aspects of the trace i.e. there occurs heterogeneity amongst the resources usage and requirement, as well as heterogeneity in duration of tasks. As per their findings large numbers of long jobs have stable resource requirements. In our research also we have concluded that there are three types of jobs prevalent in the Google cluster out of which large jobs require more resources that too for a longer duration of time.

In⁹ the authors have done a comparative study between grid and other high performance systems such as the ones collected from Grid Workload Archive and Parallel Workload ArchivewithGoogle data center. As per their study frequency of occurrence of jobs is high in Google trace and the duration of each job is also low as compared to grid systems. Besides these they have also done a study of job priorities i.e. the tasks within the same job have same priority and also a study of job lengths has been done.

In⁸ prediction of host workload is made using Bayesian modelby capturing features such as predictions and trends of data access and usage. Their results concluded that Bayes method gives high accuracy value with mean

error of 0.0014 and provides an improved load prediction accuracy value of 50% as compared to other methods.

In¹⁰ workload classification has been done by identifying workload dimensions and by using K-means algorithm to construct task classes. They have concluded that most of the tasks are short duration ones and nearly all the resources are consumed by a few tasks.

The remaining paper has been organized as follows: Section 2 gives an insight about the Google cluster trace data set. Section 3 describes the statistical analysis of trace dataset. Furthermore, Section 4 shows the clustering of jobs based on K-means++ clustering techniques. Finally the paper concludes with conclusion and future directions in Section 5.

2. Google Cluster Trace Overview

Google Cluster trace consists of a trace of about 11K machines and 700K jobs running over a span of 29 days. This is the latest version of trace released in November 2014. The cluster consists of racks of machines and each rack consists of several machines packed together. Processing takes place in cluster in form of jobs, where each job is composed of several tasks having varied resource requirements. The trace contains quite a lot of information about machine and jobcharacteristics.

2.1 Machine Events

Machine Events information is present in trace in form of timestamp; machineID of machines, event types of machine such as when machine becomes available, when a machine is removed from cluster and when a machine changes its available resources. Apart from this, information such as micro architecture and platform number along with CPU and memory capacity are also available.

2.2 Machine Attributes

Machine properties such as speed of clock and external IP address is representative of attributes in the trace.

2.3 Job Events

Job Events information is well illustrated in terms of jobs which are running or waiting and scheduling class of jobs indicating latency sensitivity of class.

2.4 Task Events

Task Events provides insight into the priority value of tasks such as free priorities, production priorities and monitoring priorities. It also contains information about the request for resources like CPU, RAM and disk usage made by each of the tasks.

2.5 Task Constraints

Each task may have many constraints associated with it. The task constraints are represented through timestamp, job id, index value, machine attribute value, name and comparison operators like Less Than, Greater Than, Equal and Not Equal.

2.6 Task Resource

The machines in Google cluster make use of Linux containers. The Task resource usage is represented in form of information like start time, end time, job ID, canonical, assigned, unmapped page cache and total page cache memory usage information. Besides these other information's such

As disk space usage, I/O time usage, cycle per instruction and memory access per unit are also present in this table.

3. Statistical Analysis of Google Trace

In order to carry out statistical analysis of the trace the data was analyzed using Apache Hive and Matlab. Figure1 showsthe workflow diagram of the statistical approach that was adopted by us in our statistical study. We first did initial data filtering using Pig, followed by feature extraction and processing through hive and lastly statistical study was done through Matlab. Apache Hive is software based on data warehouse. It provides facilities for management and analysis of large datasets on distributed storage systems such as Hadoop Distributed File System (HDFS). The use of HDFS ensures scalability and high availability of data at all times. Reading and writing of files to HDFS was conceded out through Apache HCatalog. Querying of data was carried out using HiveQL.

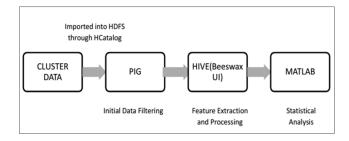
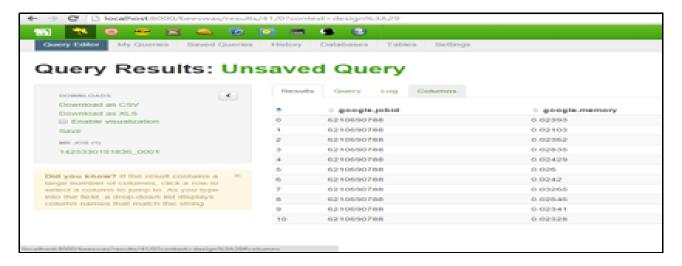
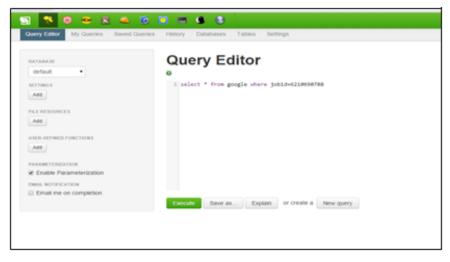


Figure 1. Workflow diagram of Statistical Analysis.

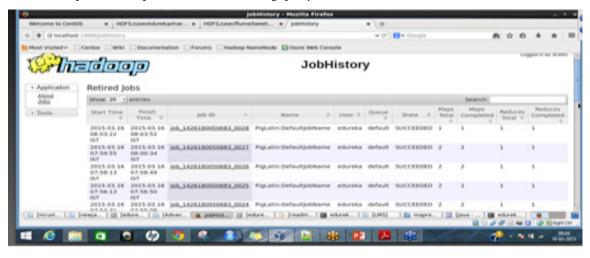
Figure 2 shows a snapshot of query to trace dataset using Hive while Figure 3. Shows that of Pig. Pig is a scripting language on top of Hadoop that enables analysis of large structured and semi structured data. FurthermoreMatlab was used for carrying out other statistical analysis of the trace. Statistical Analysis of the trace dataset was done for the purpose of analyzing the distribution of arrival times of jobs, distribution of the jobs on the basis of resource usage and also clustering of the jobs has been done using K-means++ clustering algorithm.





(b)

Figure 2. (a) Snap Shot of Hives Beeswax UI for processing and querying trace data (b) Snapshot of output screen after running query on Hive.



(a)

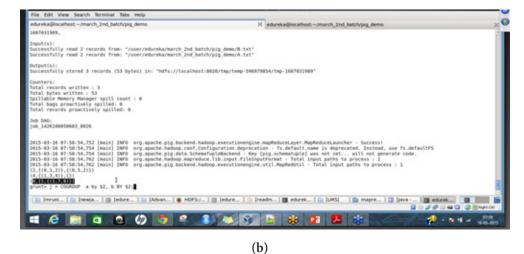


Figure 3. (a) Snap Shot of Pig grunt shell (b) Snapshot of output screen showing job history after running a Pig Script.

3.1 Clustering of Jobs based on Resource **Utilization**

In order to find out patterns amongst the jobs in cluster trace K-means++ clustering was performed. The K-means++ algorithm has significant advantages over the traditional K means algorithm which has been used for clustering of workloads in 11,12. It first finds an initial seeding value which offers considerable advantage by providing faster convergence than the traditional K-means. The K-means++ clustering algorithm defines a method for initializing cluster centers in advance preceded by the standard K-means algorithm. According to K-means++ algorithm¹³, First a center is chosen randomly from data points, then distance between each data point and its nearest center is computed. After this a new data point is chosen at random based on weighted probability distribution. The previous two steps are then repeated until the desired K centers are chosen. After this choice of initial data centers K-means clustering is performed.

Figure 4 shows the results when clustering was performed for the jobs of the trace using K-means++. From the results obtained three clear clusters of the jobs were visible and thus we could classify the jobs into three categories based on resource usage i.e. we can deduce that the resource usage of jobs in cluster trace is tri-modal in nature. This tri-modal behavior also presented in 12 has been further validated in this paper. Some jobs utilize a large amount of resources, some utilize very small number of resource while others have medium amount of resource requirements. Thus we can classify the jobs as major resource usage jobs, minor resource usage jobs and mediocre resource usage jobs. For the purpose of experimentation CPU and memory are the resources that

have been taken into consideration.

3.1.1 Major Resource Usage Jobs

Major resource usage jobs are the ones that require a lot of resources. These are the jobs that utilize majority of resources in the trace. Since these jobs are the most resource engaging ones therefore such jobs are mostly the longest running jobs in the cluster. Upon analysis of the trace it was revealed that number of such jobs is usually fewer than minor resource usage jobs but they have longer running time and are predominant in the trace and present most of the times. These are jobs that are usually involving complex computations. These jobs can be also be classified as large jobs.

3.1.2 Minor Resource Usage Jobs

Minor resource usage jobs are the ones that have minor resource requirements i.e. they require very few resources. Upon analysis of the trace it can be deduced that number of such job types is usually large. Such job types do occur frequently but their running time is usually short.Such jobs roughly constitute 75 percent of jobs in the trace. These jobs can be also be classified as small jobs.

3.1.3 Mediocre Resource Usage Jobs

Mediocre resource usage jobs are the ones whose resource requirements are not as large as the major resource usage jobs and also not as few as the minor resource usage jobs. The number of jobs required by them is in between the two. Their running time is also not very long. These jobs can be also be classified as middle jobs.

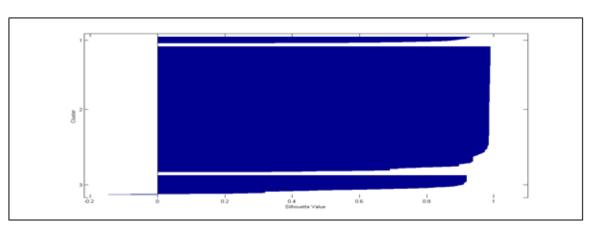


Figure 4. K-means++ clustering of jobs,

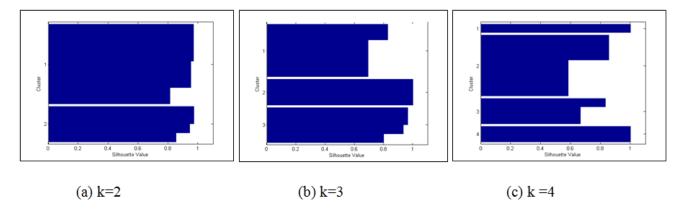


Figure 5. Clustering of Arrival times of jobs

3.2 Clustering of Arrival Time of Jobs

In order to predict the nature of arrival time of jobs clustering of jobs was done based on the arrival time of jobs using K-means++ clustering technique. Figure 5 shows silhouette results after clustering of the jobs based on arrival times. From the figure it's clear that the difference between arrival times of jobs is very less and usually several jobs arrive together in bursts. This can be attributed to the fact that frequency of querying of jobs is usually high.

3.3 Study of Distribution of Workload Parameters

3.3.1 Distribution of Arrival Time

At First distribution of arrival times of the jobs was done. This distribution was carried out through CDF where CDF refers to cumulative distribution function denoted by F(x). Cumulative density function is defined as the probability that a sample is smaller than or equal to some given

value. Equation 1^{14} shows the equation for calculating CDF

$$F(x) = \Pr(X \le x) \tag{1}$$

Where Pr(X < x) is probability that sample X is smaller than some value x.

On analysis of Figure 6 it was inferred that arrival time of jobs in Google trace roughly followed exponential

distribution but since the distribution also showed a tail we categorized them under Weibull distribution.

Also it was observed that arrival time of most of the jobs denoted by T (j_i) is less than 5(Equation 2) i.e.

$$T(j) \le 5 \tag{2}$$

Weibull distribution is defined by PDF given by Equation 3 and CDF given by Equation 4

$$F(x) = \frac{\alpha}{\beta} \left(\frac{x}{\beta} \right)^{\alpha - 1} e^{-\frac{\left(\frac{x}{\beta} \right)}{\beta} \alpha} \quad x \ge 0$$
 (3)

Where α , $\beta > 0$ and are shape and scale parameters respectively.

CDF for Weibull distribution

$$F(x) = 1 - e^{-\left(\frac{x}{\beta}\right)_{\alpha}} \quad x \ge 0 \tag{4}$$

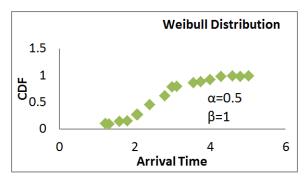


Figure 6. Weibull distribution CDF for arrival time of jobs

From study of the distribution of arrival times we can conclude that the inter arrival time between the jobs is very low.

3.3.2 Distribution of Jobs based on Resource Usage

The jobs in Google cluster have already been classified in this paper on the basis of their resource requirements. Further analysis of jobs was then carried out in order to assess distribution of jobs with respect to their resource usage.

The resources used for carrying out this analysis included CPU and memory. On analysis of the jobs on the basis of resource usage, this is further illustrated by Figure 7. It was revealed that the jobs followed a zipf-like distribution. The PDF for zipf-like distribution is given by Equation 5^{14}

$$\Pr(i) \propto \frac{1}{i^{\theta}} \tag{5}$$

The findings further revealed that some jobs required a lot of resources at all times and these jobs

are quite predominant in the cluster and therefore we need effective mechanisms for distribution of such resources and thereby more efficient scheduling algorithms are also required. Furthermore, we can deduce that large jobs usually have high average resource usage as compared to small and middle jobs. If a job is chosen at random from the cluster the probability of it being a large job is high. Thus large jobs are dominant in the cluster.

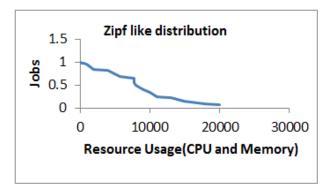


Figure 7. Zipf like distribution for resource usage

3.3.3 Distribution of Process Runtimes

On observing the runtimes of jobs, it was revealed that the distribution of runtimes of jobs is skewed with a long tail thus we could conclude that distribution of runtimes of jobs is heavy tailed in cluster trace. According to heavy tailed distribution the probability of occurrence of large values decays. Figure 8 further shows the distribution of runtimes of jobs in cluster trace. The observation of distribution of runtimes of jobs shows that some jobs had very small values of runtime while others had very large values i.e. there are many small values for runtime of jobs and comparatively fewer large values.

In heavy tailed distribution the tail decays based on power law, Equation 6^{14} .

$$\overline{F}(x) = \Pr(X > x)x^{-\alpha} \ 0 < \alpha < 2 \tag{6}$$

Where, $\overline{F}(x)$ is the survival function i.e. $\overline{F}(x) = 1 - F(x)$ and α is exponent.

The higher frequency of occurrence of shorter jobs in trace can be attributed to the fact that Google usually caters to smaller set of problems that are less time consuming such a searching for some keywords on the search engine.

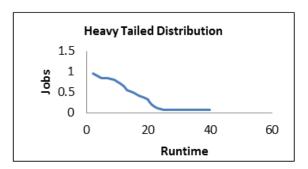


Figure 8. Heavy tailed distribution for process runtime

Study of such production workloads will help researchers in understanding production environments such as that of cloud and help in better cloud performance^{15,16}.

4. Conclusion and Future Directions

Cloud computing generally deals with non-homogeneous and dynamic environment. Google cluster trace is workload containing cell information of about 29 days spanning across 700K jobs. This paper deals with statistical analysis of Google trace. Google trace contains non-homogeneous amount of resources and their usage, study of this trace can help in making useful decisions regarding resource allocation and scheduling. In this paper we have used Hive for analysis of the trace as the size of trace is huge; use of Hive provides the advantage of

storage of data in HDFS.In this paper statistical analysis of the trace has been performed, First clustering of jobs based on resource usage has been performed and then clustering of arrival time was done. Apart from this analysis of distribution revealed several interesting results such as arrival time showed weibull distribution and inter arrival time between jobs is also very low. Distribution of jobs based on resource usage showed zipf-like distribution indicating that some jobs required a lot of resources while most of the jobs required fewer amount of resources but their frequency of occurrence is high. Finally distribution of process runtimes revealed heavy tailed distribution.

For future we have planned to expand our study to other cloud workloads such as Facebook map reduce cluster and Yahoos M45 cluster logs. We also plan to develop a Google workload cluster simulator.

5. References

- 1. Kirubakaramoorthi R, Arivazhagan, Helen D. Analysis of cloud computing technology. Indian Journal of Science and Technology. 2015 Sep; 8(21):1–3.
- 2. Shyamala K, Rani TS.An analysis on efficient resource allocation mechanisms in cloud computing. Indian Journal of Science and Technology. 2015 May; 8(9):814–21.
- 3. Alam M, Shakil KA. Recent developments in cloud based systems: State of art. 2015.
- 4. Alam M, Shakil KA. Cloud database management system architecture. UACEE International Journal of Computer Science and its Applications. 2013; 3(1):27–31.
- 5. Shakil KA, Alam M. Data management in cloud based environment using K-median clustering technique. IJCA Proceedings on 4th International IT Summit Confluence; 2013. p. 8–13.
- 6. Kavulya S, Tan J, Gandhi R, Narasimhan P. An analysis of

- traces from a production map reduce cluster. Proceedings of the 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing; Melbourne, VIC. 2010 May. p.94–103.
- 7. Abad CL, Roberts N, Lu Y, Campbell RH.A storage-centric analysis of Map reduce workloads: File popularity, temporal locality and arrival patterns.IEEE International Symposium on Workload Characterization (IISWC);La Jolla, CA.2012 Nov.p.100-9.
- Reiss C, Tumanov A, Ganger GR, Katz RH, Kozuch MA. Heterogeneity and dynamicity of clouds at scale: Google trace analysis. Proceedings of the 3rd ACM Symposium on Cloud Computing; New York, NY, USA. 2012. p.1–13
- Sheng D, Kondo D, Cirne W. Characterization and comparison of cloud versus grid workloads. IEEE International Conference on Cluster Computing (CLUSTER); Beijing. 2012. p.230–8.
- Mishra AK, Hellerstein JL, Cirne W, Das CR. Towards characterizing cloud backend workloads: Insights from Google compute clusters. ACM SIGMETRICS Performance Evaluation Review. 2010 Mar; 37(4):34–41.
- 11. Di S, Kondo D, Cappello F. Characterizing cloud applications on a google data center.42nd International Conference on Parallel Processing (ICPP); Lyon. 2013. p.468–73.
- 12. Alam M, Shakil KA, Sethi S. Analysis and clustering of workload in Google cluster trace based on resource usage. 2015.
- 13. K-means++. Available from: http://en.wikipedia.org/wiki/K-means%2B%2B, [accessed in 2015 March.9]
- 14. Feitelson DG. Workload modeling for computer systems performance evaluation. 2014.
- 15. JeonH, Min YG, Seo KK.A performance measurement framework of cloud storage services. Indian Journal of Science and Technology. 2015 Sep. 8(S8):105–11.
- Alam M, Shakil KA. A decision matrix and monitoring based framework for infrastructure performance enhancement in a cloud based environment. International Conference on Recent Trends in Communication and Computer Networks; Delhi.2013 Nov. p. 174–80.