SQUID Log Analyzer Using Hadoop Framework

Bharat Parte
Department of CSE,
Department of CSE,
VIIT, Pune, India

Umesh Jamdade
Department of CSE,
Department of CSE,
VIIT, Pune, India

Pranita Sonavane
Department of CSE,
Department of CSE,
VIIT, Pune, India

Sheetal Jadhav
Department of CSE,
Department of CSE,
VIIT, Pune, India

ABSTRACT

Squid Log Analyzer with Distributed System is a kind of Squid Log analytics software that parses log file from the server. It derives indicators about when, how and by whom a web server is visited. In today's world 80% of data captured today is unstructured from sensors used to gather climate information, posts to social media sites, digital pictures and videos, purchase transaction records and cell phones GPS signals. In Squid Log Analyzer, Log file contain information about user name, IP address, time stamp, access request, number of byte transferred, result status, URL that referred and user agent. Through Squid Log Analyzer the web log file are uploaded into the Hadoop Distributed Framework where parallel procession on log file is carried in the form of master and slaves structure. Pig scripts are written on the classified log files to satisfy certain query. The log files are maintained by the web servers. By analyzing this log files gives an idea about the user. It involves effective mining of data and also uses tools to process the log files. It also provides the idea of creating an extended log file and learning the user behavior. Analyzing the user activities is particularly useful for studying user behavior when using highly interactive systems. The main focus of our project is to build a prototype of log analyzer, studying the information-seeking process and analyzing the log files in graphical format date wise and month wise. Also information regarding hits and visiting a particular website is achieved.

Keywords

Hadoop, Log Files, Parallel Processing, Map Reduce, Pig, Hadoop Distributed File System.

1. INTRODUCTION

As per the need of today's world, everything is going online. Whenever user access internet log files get generated at server side. All of these unstructured Log data is Big Data. Hadoop is core platform for structuring Big Data and solves the problem of making it useful for analytics purposes. Hadoop can provide much needed robustness and scalability option to a distributed system. As Hadoop provides inexpensive and reliable storage and also tools for analyzing structured and unstructured data. Map/Reduce and HDFS of Hadoop use simple, robust techniques on inexpensive computer systems to deliver very high data availability and to analyze enormous amounts of information quickly. However, converting all the sequential algorithms to the parallel form, which could be converted to the map/reduce format may not be possible. There could arise situation that the algorithms may not be effectively implemented in the map/reduce format. Processing and cleansing of log files is done by Pig. Pig is a tool used to analyze large amount of data by representing them as a data flows. Using the pig lateen scripting language operations like ETL, adhoc data analysis and iterative processing can be easily achieved [3]. Pig is an abstraction over map reduce. Each and every field is having their own way of putting their applications, business online on Internet. Seating at home we can do shopping, banking related work; we get weather information, and many more services. Log files contain list of actions that have been occurred whenever someone accesses to your website or web application. These log files resides in web servers. Web Log Analyzer is a fast and powerful log analyzer [6]. It gives you information about site's visitors: activity statistics, accessed files, paths in the sites, information about referred pages, browsers, operating systems etc.[7]. The program produces easy-to-read reports that include both text information (tables) and charts each individual request is listed on a separate line in a log file, called a log entry. It is automatically created every time someone makes a request to your web site. The point of a log file is to keep track of what is happening with the web server. Log files are also used to keep track of complex systems, so that when a problem does occur, it is easy to pinpoint and fix. These log files have tons of useful information for Network Administrator, analyzing these log files can give lots of insights that help understand website traffic patterns, user activity, there interest etc[10][11]. Thus, through the log file analysis we can get the information about all the above questions as log is the record of people interaction with websites and applications.

1.1. Background

Every day huge amount of logs are generated from server. The data size is in Terabyte and thus it cannot be analyze manually. It is impossible to store and analyze these large volumes of log files. The problem of analyzing log files is complicated not only because of its volume but also because of the disparate structure of log files. Conventional database solutions are not suitable for analyzing such log files because they are not capable of handling such a large volume of logs efficiently. By comparing the SQL DBMS and Hadoop Map Reduce and suggested that Hadoop Map Reduce tunes up with the task faster. Also traditional DBMS cannot handle large datasets. This is where big data technologies come to the rescue [8]. Hadoop-Map Reduce [5] is applicable in many areas for Big Data analysis. As log files is one of the type of big data so Hadoop is the best suitable platform for storing log files and parallel implementation of Map Reduce [3] program for analyzing them. Apache Hadoop is a new way for enterprises to store and analyze data. While it can be used on a single machine, its true power lies in its ability to scale to hundreds or thousands of computers, each with several processor cores. As described by Tom White [6] in Hadoop cluster, there are thousands of nodes which store multiple blocks of log files. Hadoop is specially designed to work on large volume of information by connecting commodity computers to work I parallel. Hadoop breaks up log files into blocks and these blocks are evenly distributed over thousands of nodes in a Hadoop cluster. Also it does the replication of these blocks over the multiple nodes so as to provide features like reliability and fault tolerance. Parallel computation of Map Reduce improves performance for large log files by breaking job into number of tasks.

2. SYSTEM ARCHITECTURE

Following system architecture shown in Figure 1. Consists of major components like Squid Log File, Cloud Framework implementing Hadoop storage and Map Reduce programming model and user interface.

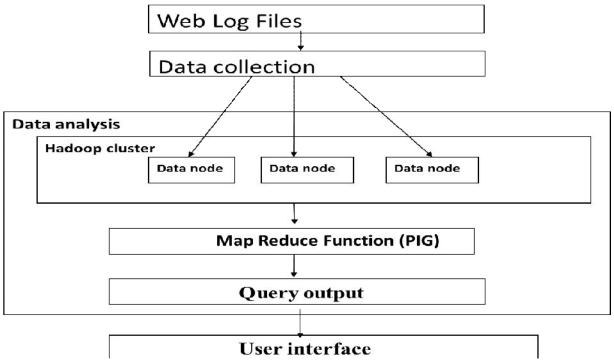

Figure 1. System Architecture

2.1. Squid Servers

Squid Server consists of multiple web servers from which log files are collected. As log files reside in a web server we need to collect them from these servers and collected log files may require pre-processing to be done before storing log files to HDFS[1][5]. Pre-processing consists on cleaning log files, removing redundancy, etc. Because we need quality of data, pre-processing has to be performed. So these servers are responsible for fetching relevant log files which are required to be processed.

2.2. Hadoop Framework

Hadoop consists of storage module and processing Map Reduce [3] model. Many virtual servers configured with Hadoop stores log files in distributed manner in HDFS [1][5]. Dividing log files into blocks of size 64MB or above we can store them on multiple virtual servers in a Hadoop cluster. Workers in the Map Reduce are assigned with Map and Reduce tasks. Workers do parallel computation of Map tasks. So it does analysis of log files in just two phases Map and Reduce wherein the Map tasks it generate intermediate results (Key, value) pairs and Reduce task provides with the summarized value for a particular key. Pig [2] installed on Hadoop virtual servers map user query to Map Reduce jobs because working out how to fit log processing into pattern of Map Reduce is challenge [10]. Evaluated results of log analysis are stored back onto virtual servers of HDFS.

2.3. User Interface

This module communicate between the user and SQUID system allowing user to interact with the system specifying processing query, evaluate results and also get visualize results of log analysis in different form of graphical reports.

3. IMPLEMENTATIONOF SQUID LOG SERVER

SQUID log processor is implemented in three phases. It includes log pre-processing, interacting with HDFS and implementation of Map Reduce programming model.

3.1. Interaction with HDFS

Hadoop framework consist of five daemons namely Namenode, Datanode, Jobtracker, Tasktracker, Secondary namenode. In pseudo distributed mode all the daemons run on local machine simulating a cluster. Hadoop, including HDFS, is well suited for distributed storage and distributed processing using commodity hardware. It is fault tolerant, scalable, and extremely simple to expand. Map-Reduce, well known for its simplicity and applicability for large set of distributed applications, is an integral part of Hadoop. HDFS is highly configurable with a default configuration well suited for many installations. Most of the time, configuration needs to be tuned only for very large clusters. It is written in Java and is supported on all major platforms. Supports shell like commands to interact with HDFS directly. Name node and Datanode have built in web servers that make it easy to check current status of the cluster. New features and improvements are regularly implemented in HDFS. The following is a subset of useful features in HDFS: File permissions and authentication.

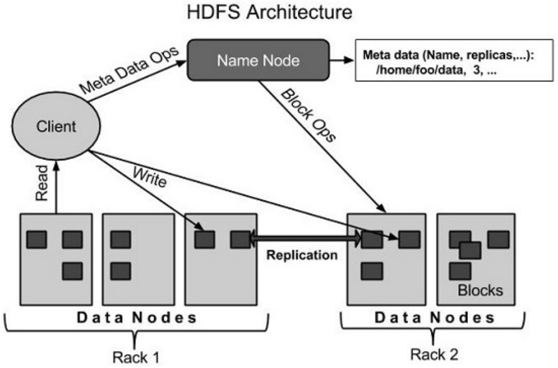

Figure 2. Hadoop Distributed File System

Hadoop Distributed File System holds a large log files in a redundant way across multiple machines to achieve high availability for parallel processing and durability during failures. It also provides high throughput access to log files. It is block-structured file system as it breaks up log files into small blocks of fixed size.

3.2. Map Reduce Framework

A Map Reduce job usually splits the input data-set into independent chunks which are processed by the map tasks in a completely parallel manner. The map, maps the job into key and value. The framework sorts the outputs of the maps, which are then input to the reduce tasks. The input of reduce and output of map must have same type. Typically both the input and the output of the job are stored in a file system. The output from the map is stored in the temporary file in the HDFC, after completion of the all the map reduce task the file is converted into the permanent one. The framework takes care of scheduling tasks, monitoring them and re-executes the failed tasks. Map Reduce does conversion twice for the two major tasks: Map and Reduce just by dividing whole workload into number of tasks and distributing them over different machines in Hadoop cluster. As we know, logs in the log files are also in the form of lists. Log file consists of thousands of records i.e. logs which are in the text format.

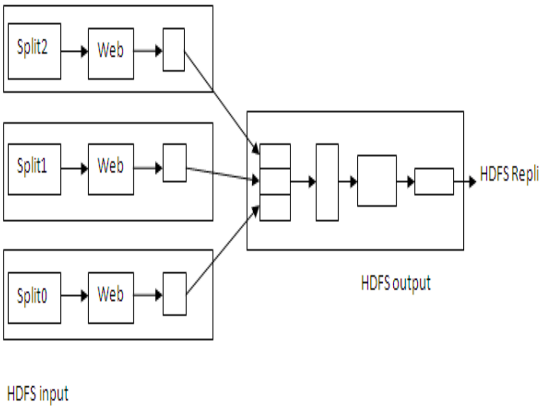

Figure 3. Architecture of Map Phase

3.2.1. Map Phase

Input to the Map Reduce is log file, each record in log file is considered as an input to a Map task. Map function takes a key-value pair as an input thus producing intermediate result in terms of key-value pair. It takes each attribute in the record as a key and Maps each value in a record to its key generating intermediate output as key-value pair. Map reads each log from simple text file, breaks each log into the sequence of keys (x1, x2, ….,xn) and emits value for each key which is always 1. If key appears n times among all records then there will be n key-value pairs (x, 1) among its output.
Map: (x1, v1) □ [(x2, v2)]

3.2.2. Reduce Phase

Reduce task takes key and its list of associated values as an input. It combines values for input key by reducing list of values as single value which is the count of occurrences of each key in the log file, thus generating output in the form of key-value pair (x, sum).
Reduce: (x2, [v2]) □ (x3, v3)

4. CONCLUSIONS

The main objective of the SQUD Log Analyzer is to submit the report on the best of Log Analyzer and help the developers to analyze reports. It gives suggestions on the improvement in huge data processing by using the Hadoop framework, which makes the data retrieval, and processing fast. Analyzing the user activities is particularly useful for studying user behaviour when using highly interactive systems. A prototype of web log analyzer is build for studying the information seeking process, finding log errors and analyzing the log files in graphical format date wise and month wise. Also the information regarding popularities and visitors visiting a particular website is achieved. Hadoop – SQUID log file analysis tool will provide us graphical reports showing hits for webpages, user's activity, in which part of website users are interested, traffic sources, etc. Hadoop MapReduce framework provides parallel distributed processing and reliable data storage for large volumes of log files. Firstly, data get stored in the hierarchy on several nodes in a cluster so that access time required can be reduced which saves much of the processing time. Here hadoop's characteristic of moving computation to the data rather Secondly, MapReduce successfully works for large datasets giving the efficient results.

REFERENCES

[1] K. Christodoulopoulos, V. Gkamas, and E.A. Varvarigos, ''Statistical Analysis and Modeling of Jobs in a Grid Environment,'' J. Grid Comput., vol. 6, no.1, 2008.

[2] J. Dean, S. Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters," In Proc. of the 6th Symposium on Operating SystemsDesign and Implementation, San Francisco CA, Dec. 2004.

[3] Jeffrey Dean and Sanjay Ghemawat., (2004) "MapReduce: Simplified Data Processing on Large Clusters", Google Research Publication.

[4]http://www.michaelnoll.com/tutorials/running-hadoop-on-

[5] Tom White, (2009) "Hadoop: The Definitive Guide. O'Reilly", Scbastopol, California.

[6] M. Zaharia, D. Borthakur, J. S. Sarma, K. Elmeleegy, S. Shenker, and I. Stoica, "Job scheduling for multi-user map reduce clusters," EECS Department, University of California, Berkeley, Tech. Rep., Apr 2009

[7] T. Hastie and R. Tibshirani. Discriminant analysis by gaussian mixtures. Journal of the Royal Statistical Society B, pages 155–176, 1996 J. Dean and S. Ghemawat. MapReduce: Simplified Data Processing Cluster",Commune.ACM,51(1):107-113,2008.

[8] C. Reiss, A. Tumanov, G.R. Ganger, R.H. Katz, and M.A. Kozuch, ''Heterogeneity and Dynamicity of Clouds at Scale: Google Trace Analysis,'' in Proc. SoCC, 2012, p. 7.

[9] K. Ren, G. Gibson, Y. Kwon, M. Balazinska, and B. Howe, ''Hadoop's Adolescence; a Comparative Workloads Analysis from Three Research Clusters,'' in Proc. SC Companion, 2012, p. 1452.

[10] Bernard J. Jansen,"The Methodology of Search Log Analysis",ubuntulinux-single-nodecluster/. Pennsylvania State University,USA.