A Novel Approach for Generic log analyser

Reshma Chaudhari[1]
Student of BE Computer Engineering
BVCOE & RI, Nasik, India
University of Pune
Reshmachaudhari1807@gmail.com

Naykar Savita Dilip[2]
Student of BE Computer Engineering
BVCOE & RI, Nasik, India
University of Pune
naykarsavita@gmail.com

Vandhan Himali Jaywant[3]
Student of BE Computer Engineering
BVCOE & RI, Nasik, India
University of Pune
Himalivadhan34@gmail.com

Shelke Pratibha Ashok[4]
Student of BE Computer Engineering
BVCOE & RI, Nasik, India
University of Pune
Shelkepratibha17@gmail.com

Prof. Kavita S. Kumavat[5]
ME Computer Engineering
BVCOE & RI, Nasik, India
University Of Pune
Kavitakumavat26@gmail.com

Abstract **-** To capture the meaning of this emerging trend the term big data was formulated. In addition to its sheer volume, big data also shows other unique characteristics as compared with traditional data. For instance, big data requires more real-time analysis and is commonly unstructured. For data acquisition, transmission, storage, and large-scale data processing components, this improvement calls for new system architectures. In all databases there are log files that keep records of changes in database. This can include tracking distinct user events. For log processing Apache Hadoop is used. A standard part of large applications are the log files and are essential in operating systems, computer networks and distributed systems. The only ways to identify and locate an error in software log files are used, because log file analysis is not affected by anytime-based issues known as probe effect. This is opposite to analysis of a running program, when the investigative process can obstruct with time-critical or resource-critical conditions within the analyzed program. The global goal of this project is to design a generic log analyzer using hadoop map-reduce framework. Different kinds of log files such as- Email logs, Web logs; Firewall logs Server logs, Call data logs are analyzed using generic log analyzer.

Keywords*- Map Reduce, Hadoop HTTP, Log analyzer, Log file.*

*****

I. INTRODUCTION

Late programming applications frequently deliver (or can be configured to create) some assistant content files known as log files. Such files are utilized amid different phases of programming improvement, essentially to debug and profiling purposes. Utilization of log files helps making so as to test troubleshooting straightforward. It permits you to take after the program's rationale, at abnormal state, need needing to run it in investigate mode. These days, log files are regularly utilized at client's establishments for the reason for lasting programming checking and/or fine-tuning. Log files turned into a standard piece of substantial application and are fundamental in working frameworks, organizing in PCs and conveyed frameworks. Log files are regularly the main route how to perceive and find a blunder in programming, in light of the fact that log file investigation is not influenced by whenever based issues known as test impact. This not likes an examination of a running system, when the diagnostic procedure can meddle with time-systematic or asset scientific conditions inside of the dissected project. Log files are regularly vast and can have troublesome structure. In spite of the fact that the procedure of making log files is very basic and straight forward, log files examination could be a gigantic errand that requires immense computational assets, long time and modern methods. This regularly prompts a typical cir ceaselessly created and possesses significant space on capacity gadgets, yet no one uses them and conveys encased data. The general objective of this undertaking is to plan a non specific log analyzer by utilizing a hadoop mapreduce system. This non specific log analyzer can break down various types of log files, for example, Email logs, Web logs, Firewall logs Server logs, shout information log

- ➢ Motivation:

The available log analyzers are able to analyze single type of log file only. To make the system capable of analyzing different kinds of log files and to make smart use of hadoop map-reduce framework to increase the efficiency is main motivation to build the system. Built system is able to analyze different kinds of log files with its increased efficiency due to use of hadoop mapreduce framework.

- ➢ Background:

The diverse sorts of log analyzers like awstats, firegen have the capacity to break down specific sorts of log files just. These log analyzers may not be efficient over circulated frameworks dissimilar to the proposed framework which makes utilization of hadoop mapreduce structure.

- Need:

There are different applications (known as log file analyzers for log files to deliver effortlessly intelligible synopsis reports. Such devices are without a doubt valuable, yet their use is restricted just to log files of certain structure. While such items representation instruments) that can process a log file of specific seller or structure and have configuration alternatives, they can answer just imbued inquiries and make mixed reports to outline an open, exceptionally flexible measured apparatus, that would be competent to break down practically. There is additionally a conviction that it is valuable to research in the field of log files examination and any log file and answers any inquiries, including extremely complex ones. Such analyzer ought to be programmable, extendable, efficient (on account of the volume of log files) and simple to use for end clients. It ought not be constrained to investigate simply log files of specific structure or sort.

II. Literature Survey

Web search engines, Intranets, and Websites can contribute important insights into understanding the information searching approaches of online searchers, by applying the data stored in search logs. The information system design, interface development, and information architecture for content collections can be informed using that understanding. This presents a review of foundation for conducting Web search transaction log analysis.

In the past decades, from log files there was unexpected low attention paid to problem of getting useful information. It seems there are two main streams of research [1]. The first one concentrates on validating program runs by checking conformity of log files to a state machine. Records in a log file are interpreted as transitions of given state machine. If some unlawful transitions occur, then there is exactly a problem, either in the software under test or in the state machine specification or in the testing software itself. Articles that just describe so many ways of production statistical outputs are represented using the second branch of research. The following item summarizes current possible usage of log files:

- Generic program debugging and profiling
- Tests whether program conforms to a given state machine different usage statistics, top ten, etc.
- Security monitoring

In operating systems Log files have become essential and became a standard part of large applications, computer networks and distributed systems. According to the available scientific papers it appears that www industry is the most evolving and developed area of log file analysis. Log files of HTTP servers are currently used not only for system load statistic but they offer a very expensive and cheap source of feedback. Providers of web content were the first one who lack more detailed and experienced reports based on server logs. They require detecting behavioural patterns, paths, trends etc. Simple statistical methods do not fulfil these needs so an advanced approach must be used [2].

Log files are having complicated structure and are often very large. Although, log files generating process is quite uncomplicated and straightforward, log file analysis could be a enormous task. There are over 30 commercially available applications for web log analysis and many so many are free available on the internet. Regardless of their price, they are disliked by their user and considered too low, inflexible and difficult to maintain. Some log files, especially small and simple, can be also analyzed using familiar spreadsheet or database programs. In such case, the logs are imported into a worksheet or database and then analyzed by using accessible functions and tools [3]. The remaining but probably the most promising way of log file processing represents data driven tools like AWK. In connection with regular expressions are such tools very efficient and flexible. On the other hand, they are too low-level, i.e. their usability is limited to text files, one-way, single-pass operation. Or higher-level tasks they lack mainly advanced data structures. Some Currently available log analysers are as follows:

- Server log analyser eg.AWStats
- Firewall log Analyzer eg. Firegen
- Call Data Analyzer eg. aMiner

Current Log analyzers are able to analyze particular log files only. Eg. Capsa a network analyzer is able to analyze single type of log file such as http log file.

III. System Overview:

Problem Statement

To build a system for generic log analysis using Hadoop Map Reduce framework by providing user to analyze different types of large-scale log files.

- Hadoop

Hadoop is a software framework that supports data-intensive distributed applications under a free license. It allows applications to work with thousands of computational independent computers and petabytes of data. Hadoop was extracted from Google’s Mapreduce and Google File System (GFS) papers. Hadoop is a top-level Apache project being built and used by an international community of contributors, written in the Java programming language. Yahoo! has been the biggest contributor to the project, and uses Hadoop extensively across its businesses. Hadoop is buildup of mainly two parts:

1. HDFS
2. Map-Reduce Framework

1. HDFS-

Fig1 shows HDFS architecture. HDFS is the primary distributed storage used by Hadoop applications. A HDFS cluster primarily made up of a NameNode that manages the file system metadata and Datanodes that store the actual data. The architecture of HDFS is described in outfit here. This user guide primarily deals with interaction of users and administrators with HDFS group. The diagram from HDFS architecture depicts basic interactions among NameNode, Datanodes, and the clients. Essentially, clients contact NameNode for file metadata or file modifications and perform actual file I/O directly with the Datanodes. The following are some of the salient features that could be of interest to many users. The terms in italics are explained in later segments. Hadoop, including HDFS, is well suited for distributed storage and distributed processing using product hardware. Supports shell like commands to interact with HDFS directly. NameNode and Datanodes have fixed web servers that make it easy to check present status of the cluster.

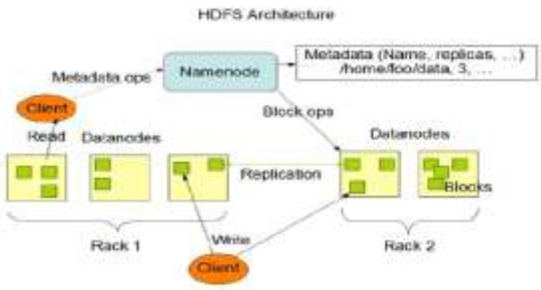

Fig.1 HDFS Architecture

New features and improvements are usually implemented in HDFS. The following is a subset of useful appearance in HDFS:

• File permissions and attestation.

• Rack awareness: to take a node's natural location into account while scheduling tasks and allocating storage.

• Safe mode: an administrative mode for maintenance.

• Sck: a utility to diagnose health of the filesystem, to find missing files or blocks.

• Rebalancer: tool to balance the cluster when the data is unevenly distributed among Datanodes

• Upgrade and Rollback: later a software boost, it is possible to rollback to HDFS' state before the upgrade in case of accidental problems.

• Secondary NameNode: advices to keep the size of file containing log of HDFS modification with in certain limit at the NameNode.

2. Map Reduce

Fig2. Shows MapReduce data flow. Hadoop Map-Reduce is a software framework for easily writing applications which process large amounts of data (multi-terabyte data-sets) in-parallel on large clusters (thousands of nodes) of commodity hardware in a decisive, fault-tolerant manner. A Map-Reduce job usually splits the input data-set into independent chunks which are processed by the map projects in a completely parallel manner. The framework sorts the outputs of the maps, which are then input to the reduce project. This configuration allows the framework to effectively schedule tasks on the vertices where data is already present, resulting in very high combined bandwidth across the cluster. The Map-Reduce framework consists of a sole master Job Tracker and one slave Task Tracker per cluster-node.

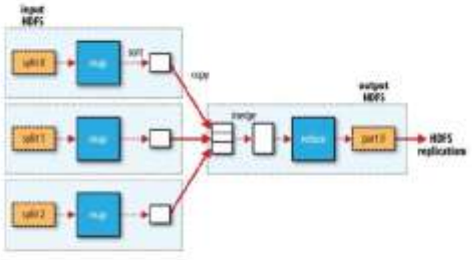

Fig 3.2 Map-reduce data flow

Fig.2 MapReduce Data Flow

The master is answerable for scheduling the jobs' component work on the slaves, monitoring them and re-executing the failed tasks. The slaves finish the tasks as directed by the master. The log file may have extensions like .txt or .log. Minimally, applications specify the input/output locations and supply map and reduce functions through implementations of suitable interfaces and/or abstract-classes. These, and other job parameters, comprise the job configuration. The Hadoop job client then agree the job (jar/executable etc.) and configuration to the JobTracker which then considers the responsibility of distributing the software/configuration to the slaves, scheduling tasks and monitoring them, contributing status and diagnostic information to the job-client.

METHODS

Although the purpose and application concerns differ, some common methods are useful for almost all of the analysis. Below, we deliberate three types of data analysis methods.

• Data visualization: Data Visualisation is closely relevant to information graphics and information visualization. To communicate information clearly and effectively through graphical means, the goal of data visualization is used. In usual, charts and maps helps people to understand information easily and quickly. However, traditional spreadsheets cannot handle the huge volume of data, as the data volume goes on increasing to the level of big data. Visualization for big data has become an alive research area because it can assist in algorithm design, software evolution, and customer engagement. Friedman and Frits summarized this field from the information representation and computer science perspectives, resultantly [1].

• Statistical analysis: is based on statistical hypothesis, which is a branch of applied mathematics. Within statistical hypothesis, randomness and uncertainty are modeled by probability theory. Statistical analysis can serve two purposes for large data sets: description and conclusion. Illustrative statistical analysis can summarize or describe a collection of data, whereas inferential statistical survey can be used to draw inferences about the process [2].

• Data mining: Is the computational activity of discovering patterns in large data sets. Various data mining algorithms have been developed in the artificial intelligence, machine learning, pattern recognition, statistics, and database areas. On Data Mining (ICDM), during the 2006 IEEE International Conference was held, the ten most influential data mining algorithms were identified based on rigorous election. In ranked order, the algorithms are C4.5, k-means, SVM (Support Vector Machine), a deduced, AdaBoost, kNN, EM (Expectation Maximization), PageRank, Naive Bayes, and CART.

IV. Algorithms

1) Job Scheduling:

Early versions of Hadoop had a very uncomplicated approach to scheduling users' jobs: they ran in order of submission, adopting a FIFO scheduler. Typically each job would use the whole cluster, so jobs had to wait their turn.

Although a shared cluster attempts a great potential for offering large resources to many users, the problem of sharing resources fairly between users desires a better scheduler. Production jobs need to complete in a timely manner, while allowing users who are creating smaller ad hoc queries to get results back in a reasonable time. Later on, the capability to set a job's priority was combined, via the mapred.job.priority property or the setJobPriority () process on Job Client (both of which take one of the characters VERY_HIGH, HIGH, NORMAL, LOW, VERY_LOW). When the job scheduler is selecting the next job to run, it selects one with the chief priority. However, with the FIFO scheduler, priorities do not support preemption, so a large-priority job can still be blocked by a long-running low priority job that started before the large-priority job was anticipated. Mapreduce in Hadoop now comes with a choice of schedulers. The default is the real FIFO queue-based scheduler, and there also a multi-user multicipator called the Fair Scheduler.

Conclusion:

At present, for all real domain problems such as web log analysis, fraud detection and text analysis the big data technology is incorporated successfully. Efficiency of log analysis has improved due to the use of Hadoop framework. It will be easy to extend our project to analyze that log file, if any new standard format log file is created. Data generation, data storage, data acquisition and data analysis are the four stages of big data value chain. In the big data generation phase, we have listed various potentially rich big data sources and discussed the data attributes. In the big data storage phase, various cloud based NoSQL stores were introduced, and several key features were compared to assist in big data design judgments. Also, we introduced the backbone of the big data movement, Hadoop, and mapreduce framework [1].

References

[1] Bernard J. Jansen, "The Methodology of Search Log Analysis", Pennsylvania State University, USA.
[2] J.H. Andrews, "Theory and Practice of Log File Analysis Technical Report", Pennsylvania Western Ontario.
[3] Jan Waldamn,"Log File Analysis Technical Report".

Chaudhari Reshma she is Engineering student of Information Technology at Brahma Valley College of Engineering And Research Institute, Nasik under University of Pune. Her interest in the field of security.

Naykar Savita Dilip she is student of Engineering student of Information Technology at Brahma Valley College of Engineering And Research Institute, Nasik under University of Pune. Her interest is in the field of security and networking.

Vandhan Himali Jaywant
She is student of Engineering student of Information Technology at Brahma Valley College of Engineering And Research Institute, Nasik under University of Pune. Her interest in the field of security.

Shelke Pratibha Ashok.
She is student of Engineering student of Information Technology at Brahma Valley College of Engineering And Research Institute, Nasik under University of Pune. Her interest in the field of security

K. S. Kumavat, ME, BE Computer Engg. Was educated at Pune University. Presently she is working as Head Information Technology Department of Brahma Valley College of Engineering and Research Institute, Nasik, Maharashtra, India. She has presented papers at National and International conferences and also published papers in National and International Journals on various aspects of Computer Engineering and Networks. Her areas of interest include Computer Networks Security and Advance Database.