International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 04 Issue: 05 | May -2017 www.irjet.net p-ISSN: 2395-0072

A Specialized Log Analysis Engine in Distributed Environment

Abhiruchi Shinde[1], Neha Vautre[2], Prajakta Yadav[3] , Sapna Kumari[4]

[1,2,3,4]*Dept of Computer Engineering, SITS, Maharashtra, India*

Abstract - *Log file or logs in computing are the files for keeping record of the events that occur in the operating system or communication between the users or operating systems. Log files contains large amount of valuable information about the system operation status, usage, user behavior analysis etc. Due to extensive use of digital appliances in today's modern era log file analysis has become a necessary task to track system operation or user behavior and acquire important knowledge based on it. These kinds of files are generated at stupendous rate and to analyze them is tedious task and a burden to corporations and various organizations. In order to analyze large dataset, and to store it efficiently, economically and effectively we need to have an effective solution which needs not only the massive and stable data processing ability but also the adaptation to a variety of scenarios under the requirement of efficiency. Such capabilities can't be achieved from standalone analysis tools or even single cloud computing framework. The main objective of the proposed system is to design an application for log analysis and applying the data mining algorithm to get the results which will be useful for system administrator to take proper decisions. The combination of Hadoop, Spark and the data warehouse and analysis tools of Hive and Shark makes it possible to provide a unified platform with batch analysis and in-memory computing capacity in order to process log in a high available, stable and efficient way. Statistics based on customer feedback data from the system will help in greater expansion of business and a company that will have such data to its disposal and ready to use in the distributed environment for log analysis*

Key Words: **Log, Weblog, Hadoop, Spark, Log analysis.**

1. INTRODUCTION

Big data analytics is the process of examining huge amount of data present in structured or unstructured form generated at a high speed to uncover hidden patterns, unknown correlations, market trends, customer preferences and other useful business information. The analytical findings can lead to more effective marketing, new revenue opportunities, better customer service, improved operational efficiency, competitive advantages over rival organizations and other business benefits. . With big data analytics, data scientists and others can analyze huge volumes of data that conventional analytics and business intelligence solutions can't touch. Log analysis is the way to gather information of the number of access users, user behavior, operation status etc. There have been some free powerful log analysis tools like Awstats , Webalizer , and Google Analytics. But they are either standalone or have the limitation of data scale. Consequently the higher number of log producing on a daily basis has led to the analysis task to be more hectic and tedious, in large databases, so we here propose a system for log analysis. In this system the web server logs are analyzed in distributed environment. Due to digitization in today's modern era log file analysis has become a necessary task to track system operation or user behavior. Log analysis is necessary for any organization to determine how well their website is performing as marketing tool. Log analysis is a tedious task. There is a need of an effective solution for integration and parallel processing of data. We will present an engine in distributed environment for log analysis using the integration of hadoop and hive and also spark and shark for the purpose. The main objective of the project is to produce such a engine that reduces the tedious and alone machine introducing the distributed environment also reducing the overall time required for the computation to take place.

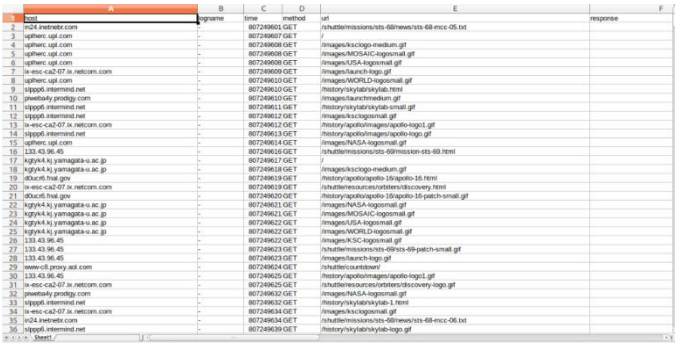

Fig1: Screenshot of web server logs

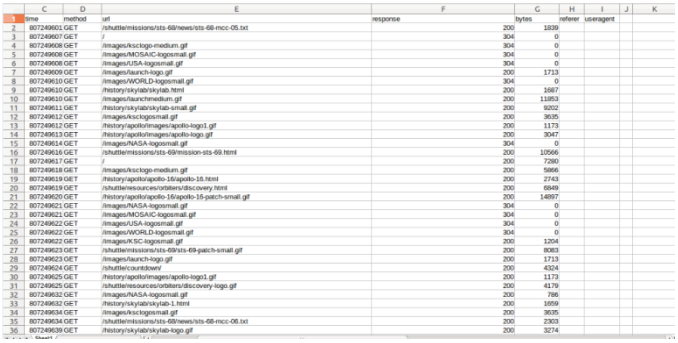

Fig.2: Screenshot of web server logs

2. EXISTING SYSTEM

There are many systems already available satisfying similar purposes. They are Webalizer and Awstats. We briefly understand the two existing technologies.
Webalizer:- The Webalizer is an application tool that generates web pages by analyzing logs, from access and usage logs, i.e. it is a software for analysis of web server logs. It is one of the most commonly used web server administration tools. It was initiated by Bradford Barret in 1997. Statistics commonly reported by Webalizer include hits, visits, referrers, the visitors' countries, and the amount of data downloaded. These statistics can be viewed graphically and presented by different time frames, such as by day, hour, or month.

3. PROPOSED SYSTEM

The previous systems as known earlier consists of the standalone machines, owing to which, there occurs a huge problem of data scalability. To address this issue and various other limitations of the previous systems, we propose to add distributed environment to the existing system. Adding this feature we try to remove the bugs of the previously proposed systems and make them work more efficiently as well as effectively, to produce desired results to the client to maintain a specific position in the market. We use map reduce operations along with the Hadoop system in HDFS to perform operations like knowing the number of hits for a particular URL, error detection, number of bytes transmitted, number of hits for each IP address. Descriptions of the entire map reduce phases along with the system architecture and results of the same are given below.

4. SYSTEM ARCHITECTURE

In any analytical tool, pre-processing is necessary, because Log file may contain noisy & ambiguous data which may affect result of analysis process. Log pre-processing is an important step to filter and organize only appropriate information before applying Map Reduce algorithm. Preprocessing reduces size of log file also it increases quality of available data. The purpose of log pre-processing is to improve log quality and increase accuracy of results.

4.1. MAP REDUCE FRAMEWORK

Map Reduce is a simple programming model for parallel processing of large volume of data.. Fundamental concept of Map Reduce is to transform lists of input data to lists of output data. Map Reduce does the 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 the Hadoop cluster. For the application which require tedious task of log data analysis, Map Reduce implementation in Hadoop is one of the best solutions. Map Reduce is divided into two phases: Map phase and Reduce phase.

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.

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

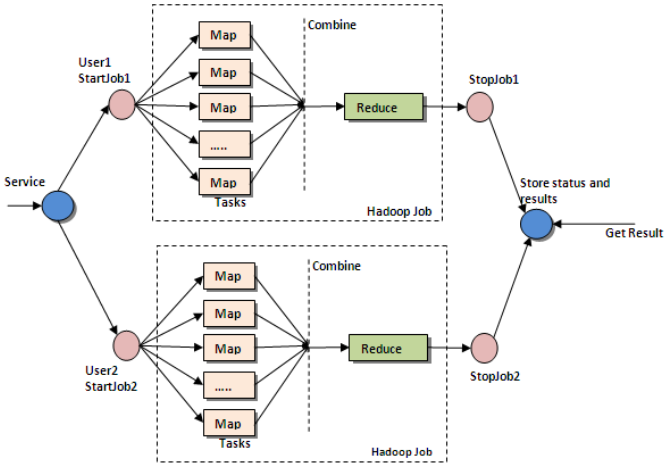

Fig.3: Map Reduce operation

The proposed system consists of major components like Web servers, implementing Hadoop storage and Map Reduce programming model and user interface.

The basic Log analysis flow in our system starts with the log data sets collection, and then ETL processing is done on log data to make it suitable for processing. Log analysis is done on hadoop as well as spark using the analytical tools hive and shark respectively. Spark does not have it's own file system so it will store it's data in HDFS. Parallel processing will be done on hadoop and for iterative query processing on shark. The result of log analysis can be given as an input to the tool named zeppeline which will show the statistical analysis of result in the form of chart.

Data cleaning is the first phase carried out in the proposed work as a pre-processing step in NASA web server log files. The NASA log file contains a number of records that corresponds to automatic requests originated by web robots, that includes a large amount of erroneous,

misleading, and incomplete information. The entries that have status of "error" or "failure" have been removed. The foremost important task carried out in data cleaning is the identification of status code. All the log lines even though satisfy the above constraints, only the lines holding the status code value of "200" is identified as correct log. The corresponding fields containing the correct status code are forwarded to the output text file. After applying data cleaning step, applicable resources are stored in the HDFS as text file, and feed back to the session Identification algorithm as input file. The cleaned web log data is used for further analysis of session identification utilized by the NASA users and also to identify unique user, unique URLs accessed. The session identification plays a major role in web log mining. The processed session file also resides in the HDFS that can be downloaded for further analysis. The pre-processed log file is used to find the user identification, as Map reduce in general identifies the unique values based on key value pair. The log file consists of different fields and the user is identified based on IP address, which is considered as key and their corresponding count as value. Once all the keys are found, the combiner is used to combine all the values of a specific key, the aggregated result is passed to the reduce task and total counts of IP accessed is listed as text file in HDFS. Other than the identification of unique user, unique fields of date, URL referred, and status code is also identified. These unique values is retrieved and used for further analysis in order to find the total URL referred on a particular date or the maximum status code got successes on specific date.

The following diagram describes

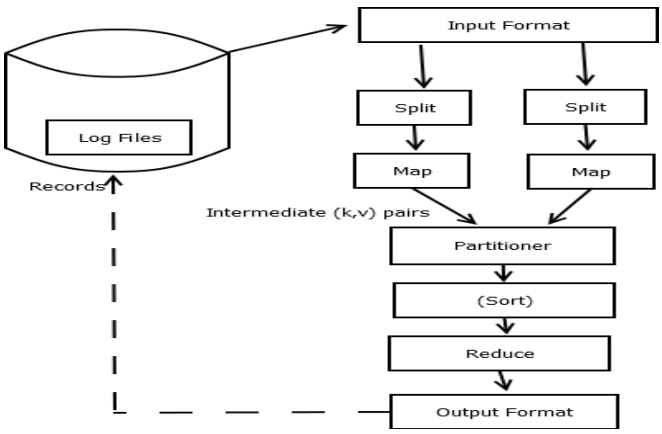

Fig.4: **System architecture**

5. RESULTS

Results in the form of screen shots are given below for number of bytes transmitted, error detection, number of hits for each URL, number of hits for each IP address. The screenshots are taken directly from the system for convenience.

Various functions are used for detecting the different types of errors that occur during transmission of data. Error messages are displayed in accordance to each error as it is predefined in the source code. For example 200 displays OK message.

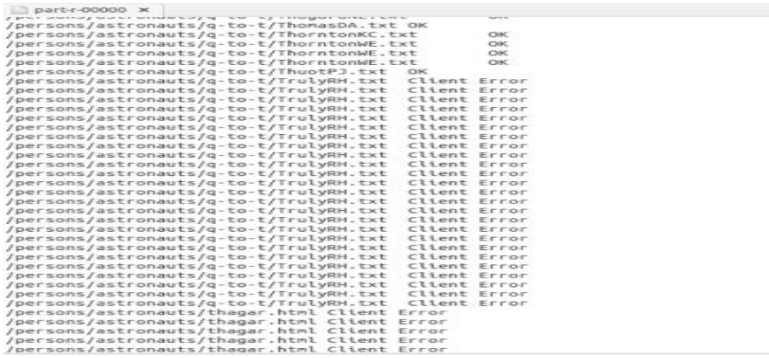

Fig.6: Screenshot for Error detection

Functionalities are used for detecting the number of bytes transmitted for each URL.

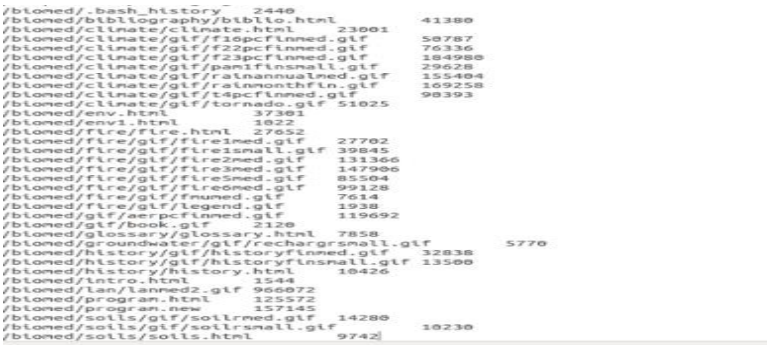

Fig.7: Number of bytes transmitted

Functionalities are used to compute the URI Hit Count for each of the URI which specifies the number of times a URI is requested. A sample of some URIs and their hit count is shown.The sorted result, as depicted shows the topmost hit URIs of the server. Thus, this analysis may be very effective in recommendation systems, web usage mining, actionable alert system and many other applications.

Fig.8: Number of hits for each URL

Counting the number of times a particular [P or host requesting the server is one of the crucial tasks in web log analysis and could be significantly useful in host centric recommendation systems and in analyzing and identifyingpotential security attackers.

Fig.9: Number of hits for each IP address

6. DISCUSSIONS

We here discuss about the various methods about the flow of log analysis and the phases for completion of the proposed system. Following is the state diagram for the proposed system.

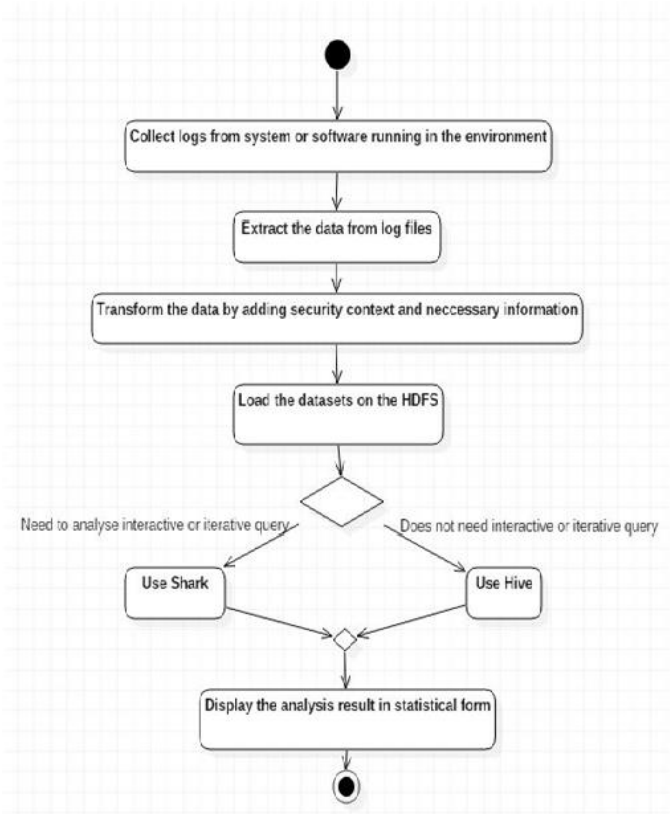

Fig.5: State diagram

7. CONCLUSION

Log analysis is the process to gather information of the number of access users, user behavior, and system operation status, etc. We design and implement a lightweight distributed framework, consisting of a minimized set of components. The framework is different from the general ones and is specially designed for log analysis of web server logs. This paper analyzes and compares the respective characteristics of Hadoop and Spark framework and Hive/ Shark. Combining the characteristics that are useful to us of Hadoop as well as Spark we propose a cloud platform analysis model with high stability, availability and efficiency for batch data analysis in comparison with standalone log analysis tools and system simply based on Hadoop or the combination of Hive.

REFERENCES

[1] Y. Q. Wei, G. G. Zhou, D. Xu, Y. Chen, "Design of the Web Log Analysis System Based on Hadoop", Advanced Materials Research, Vols. 926-930, pp. 2474-2477, 2014

[2] Xiuqin LIN, Peng WANG, Bin WU “LOG ANALYSIS IN CLOUD COMPUTING ENVIRONMENT WITH HADOOP AND SPARK”, Beijing University of Posts and Telecommunications, Beijing 100876.

[3] Xiaokui Shu, John Smiy 2013.Massive Distributed and Parallel Log Analysis For Organizational Security Industry/University Cooperative Research Center (I/UCRC), and NSF grant CAREER CNS0953638

[4] Yanish Pradhananga, Shridevi Karande, Chandraprakash Karande, "CBA: Cloud-based Bigdata Analytics",978-1-4799-6892-3/15 $31.00 © 2015 IEEE DOI 10.1109/ICCUBEA.2015.18

[5] Markku Hinkka, Teemu Leht, Keijo Heljanko, "Assessing Big Data SQL Frameworks for Analyzing Event Logs",978-1-4673-8776-7/16 $31.00 © 2016 IEEE DOI 10.1109/PDP.2016.26

[6] Jaladi Udaya Santhi, Sivaiah Bellamkonda, Dr.N.Gnaneswara Rao, "Analysis of web server log files using Hadoop MapReduce to preprocess the log files and to explore the session identification and network anomalies", 3rd International Conference on Electrical, Electronics, Engineering Trends, Communication, Optimization and Sciences (EEECOS)-2016, 2016.