International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 4 Issue 10, October-2015

Data Analytics on E-Commerce Transaction Logs for Payment Management

Vignesh S
Payments Engineering,
PayPal Inc.
Bangalore, India

Srinath N K
Dean, Computer Science & Engineering,
R V College of Engineering
Bangalore, India

Sandeep R V
Payments Engineering,
PayPal Inc.
Bangalore, India

***Abstract*—In this paper, we present an end to end distributed system for aggregation, extraction and analyze large volume of e-commerce transaction logs. The system aggregates the logs from different sources, cleans it and parses it to extract the required information. The cleaned textual data is then exported to a data warehouse like Hive for further analysis. Techniques like six sigma deviation are applied to the dataset to provide a statistical picture of the overall transaction success rate. Clustering and classification algorithms are also applied to identify various factors that influence a transaction to succeed, also relations between many such transactions can be used to narrow down the factors that influence a transaction to get declined.**

Keywords— Transaction logs; Log analysis; Big Data;

I. INTRODUCTION

In recent years, there has been an exponential growth in the amount of data generated. Big data refers to a collection of data sets so large and complex that it becomes difficult to process using traditional data processing applications. More than 80% of all potentially useful information is unstructured data, in the form of sensor readings, console logs and so on [1]. Log processing has become a critical component in the field of data analytics. It contains valuable information about the execution of a system; this could be used for debugging, operational profiling, finding anomalies, detecting security threats, measuring performance and visualizing trends [2].

Exponential data growth makes it continuously difficult to collect, store, process, and analyze complex unstructured data [3]. The proposed system aims to solve this challenge by making use of Hadoop. Hadoop is a framework that allows distributed processing of large data sets across clusters of computers using simple programming models. It is designed to scale up from a single server to thousands of machines, each offering local computation and storage. Rather than relying on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, thus delivering a highly-available service on top of a cluster of computers, each of which may be prone to failures [1].

Hadoop [4] [5] uses the concept of Map Reduce which enables to process a massive volume of data in parallel with many low-end computing nodes in the Hadoop cluster. The map-reduce paradigm is too low-level and extremely rigid because of its one-input two-stage data flow. To perform tasks having a different data flow inelegant workarounds have to be devised. Apache Pig is fully implemented on top of Hadoop's map-reduce paradigm and compiles Pig Latin into physical plans that are executed over a map-reduce implementation [6].

Hive is a data warehousing solution on top of the Hadoop MapReduce framework that has been designed to handle large amounts of data and store them in tables like a relational database management system or a conventional data warehouse while using the parallelization and batch processing functionalities of the Hadoop map-reduce framework to speed up the execution of queries [7] [8]. Data inserted into Hive is directly stored in the Hadoop Distributed File System (HDFS),

The proposed system aggregates logs generated during e-commerce payment transactions, from heterogeneous, distributed systems into a single cluster. Aggregated logs are analyzed to provide meaningful insights related to end user transactions, which includes information such as the number of cross-border transactions, trends that lead to transaction failures and information of the parameters responsible for the failure of such transactions. An experimental system was developed and applied in PayPal. PayPal does billions of transactions every day and thus was the best platform to test the system. The information shared in the paper is in public domain and certain sensitive information has been withheld while projecting the results.

II. SYSTEM ARCHITECTURE

In Fig.1, we present our system architecture. The system follows a plug and play architecture. It is divided into various modules/components that are independent of each other and the working of each component is abstract to others. A component is only bothered about the data flow from its upstream component and passing it on to its downstream component.

Logs from various sources (unstructured format) are aggregated to HDFS with the help of Apache Flume. It provides an option to configure multi-tier system which is managed by configuring a certain number of agents at each tier.

The raw unstructured data is parsed using Apache Pig to eliminate irregularities in the records. The preprocessed data is stored into a Data Warehouse called Hive [7] [8].

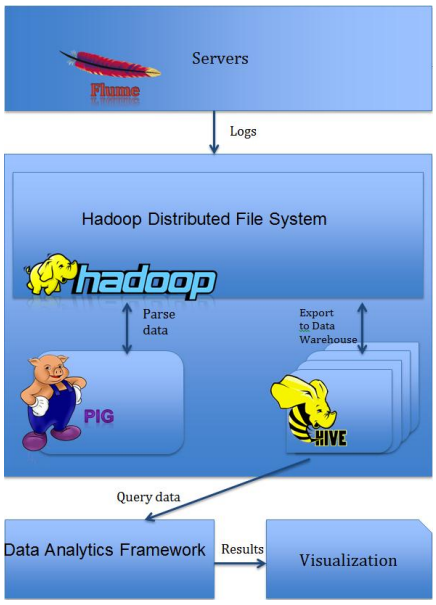

Figure 1. System Architecture

Depending on the use case, suitable techniques, which involve Statistical or Machine learning models, are applied. The results of the analysis are visualized using Tableau or D3 JS.

III. IMPLEMENTATION

Transaction logs are produced by a logging library running on different servers. The rolling period of the logs is set to create log file on an hourly basis. The flume agent (A JVM instance of Apache Flume) transfers the logs files with the help of a cron job. The cron job is set to run say every 5th minute of the hour where the cron job gets the path of the logs for the previous hour and initiates the flume agent which transfers this log file.

After aggregating the log files, the unstructured log files must be converted to a structured format with the removal of data that is irrelevant. A typical log file contains data for many events during a transaction .The events of no concern have to be filtered and information has to extracted from this data. The transaction logs of PayPal have a Name Value Pair (NVP) payload. During Preprocessing, The payload is split on ampersand (&) and later key value pairs are separated on equal's (=) sign. For e.g. ID=123asds&buyer_country=US. The kinds of preprocessing include:

- Filtering the NULL
- Filtering the empty values
- Filtering the irrelevant parameters
- Filtering / merging the deprecated fields are also done.

The preprocessing module is made very generic, so that only changes made to a configuration file would be enough to clean a different dataset. The processed data is now stored into the HDFS with the help of Hive. Information from the data warehouse is then communicated to the analytics framework.

In the statistical analysis module, the conversion rate of the transactions is analyzed by selecting certain parameters like buyer country, seller country, merchant details, API version, etc. As shown in Figure 2, Data from Hive is chosen according to the selected parameters and is given to the analyzer. The analyzer uses techniques like six sigma deviation to find the outliers. Six sigma deviation is a methodology used to detect outliers from the mean. Mathematically it can be determined using the formula,

$$\text{six sigma deviation} = \bar{x} + 6 * \sigma \quad (1)$$

First the mean of the dataset is calculated and the deviation of the values from the mean are calculated. A deviation of values +-3 times (sigma) from the mean would be the normal values, anything outside this range would be classified as an outlier. The identified outliers are sent to visualizer for reporting.

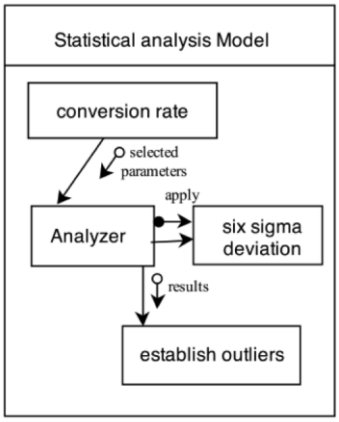

Figure 2. Statistical analysis model

The data from Hive is also sent to the Machine Learning Module. The conventional classification algorithms cannot be applied to the data set as number of fields in the dataset is very high. These algorithms use centroid based approach to form clusters and since our dataset contains many columns or fields, finding the centroid and forming clusters based on the centroids will not yield in a fruitful classification. Decision making algorithms like random forest are applied on the data set to correlate the importance of all the parameters involved in a transaction. Further analysis could be carried out on the key parameters identified in the previous iteration rather than on the whole set of parameters (nearly hundreds) associated with a transaction. This helps to scale up the system to huge dataset and infer critical decisions. The identified patterns are sent to the visualizer for reporting.

Visualization of the insights and patterns, obtained for the analysis is done using two frameworks as shown in Figure 3, Tableau and D3 JS [9]. Tableau is commercial software having both public (free version) and a paid version. The framework provides quick and easy ways to visualize the data. Charts like the Bubble chart Bar graph etc. can be create with ease using Tablue.D3 JS is open source tool for building interactive charts and graphs [9]. Interactive Charts like Pie charts with selectable parameters and bar graph with scrollable time frame can be built with D3 JS with ease. Such interactive Charts give a better understanding to infer the patters and outliers in an effective way.

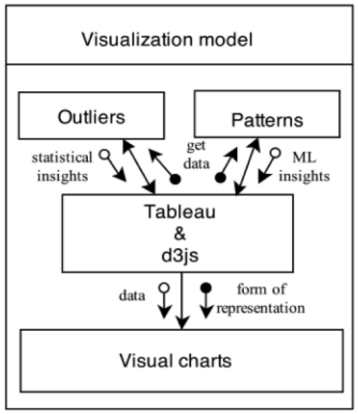

Figure 3. Visualization model

IV. Experimental Results

A. Performance Metrics

The Log Parser Module was given datasets of different sizes and the time taken by the module to parse the logs using Pig were recoded table 1.

TABLE I. Performance metrics of Log Parser

Duration	Performance metrics of Log Parser		
	Log size in MB(compressed)	*Number of Records*	*Time Taken*
1 Hour log	378	Nearly 100,000	< 2 min
1 Day log	30,000	Nearly 1,01,50,000	< 5 min

Log Analysis was done using frameworks like Pig, Hive and R for different kinds of use cases. The operations done using Hive and the statistical analyser were completed on an average less than 15 minutes for complex query based analysis.

The machine learning module took nearly 4 minutes to perform analysis a data of nearly 100 records. The module seemed to have totally frozen when it was used for analyzing data more 1000 records.

B. Inference from the Results

The above implementation methodology was executed for 15 days data. The mean of the conversion rate was calculated and all the groups in which the conversion rate was less than six sigma times the mean conversion rate were analyzed in detail. The following were the major insights got from the data:

- The list of countries with lower conversion rate was extracted.
- The merchants for which the conversion rates were less were also listed and the PPAPI (PayPal API) error codes for those merchants were seen.
- It was also observed that for many merchants, if the conversion rate was less, then there was one particular error code which occurred much more than the other error codes, meaning most of the failures can be attributed to that particular PPAPI error code. The root cause of the error was easily identified and the errors were fixed.
- From the machine learning module it was observed that the buyer country does not have much effect on the conversion rate for a particular merchant. Thus for a particular merchant which has a lower success rate, then it generally has lower success rates in all the buyer countries irrespective of the country.

From Figure 4 the API versions with the least conversion rate were determined. Then, the most common PPAPI error codes for these API versions were observed through further analysis and were visualized using a bubble chart.

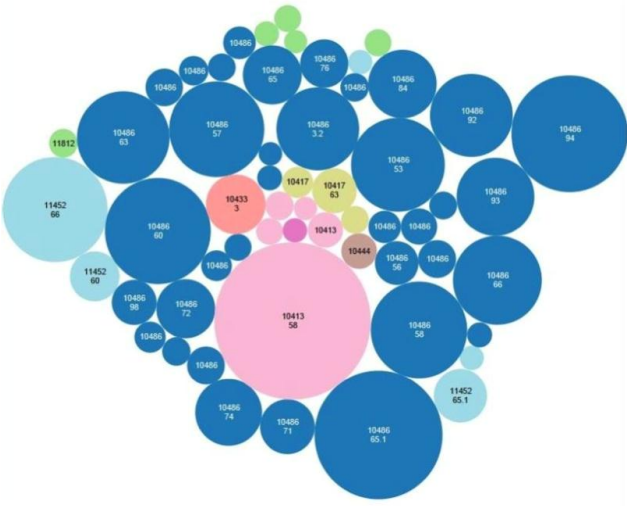

Figure 4. API Versions with popular error codes

V. Conclusion

The goal in obtaining valuable insights from the logs was achieved along with integration of a log aggregator. Map-reduce paradigm used in individual process of parsing the logs and analyzing the logs helped reducing the computational time drastically in spite of the enormous input data set. The Statistical analysis of the data was completed and certain insights were given. Using those insights a number of patterns were drawn as shown in the results. The machine learning analysis of the data gave deeper insights into the factors influencing the status or the result of an e-commerce transaction.

REFERENCES

[1] Das T.K, Mohan Kumar P, "BIG Data Analytics: A Framework for Unstructured Data Analysis," *International Journal of Engineering and Technology (IJET)*, vol. 5 Feb-Mar 2013, pp.1153-156.

[2] Nagappan M, Vouk M.A, "Abstracting log lines to log event types for mining software system logs," Mining Software Repositories (MSR), *2010 7th IEEE Working Conference on* 2-3 May 2010, pp.114,117.

[3] Alfredo Cuzzocrea , Il-Yeol Song , Karen C. Davis, "Analytics over large-scale multidimensional data: the big data revolution!", *at Glasgow, Scotland, UK proceedings of the ACM 14th international workshop on Data Warehousing and OLAP*, October 28-28, 2011,pp.101,104

[4] White T., Hadoop: The Definitive Guide. O'Reilly Media, Yahoo! Press, June 5, 2009.

[5] Venner J., ProHadoop, Apress on June22, 2009.

[6] Sanjeev Dhawan , Sanjay Rathee, "Big Data Analytics using Hadoop Components like Pig and Hive" *American International Journal of Research in Science, Technology, Engineering & Mathematics*, 2013, pp. 1,5.

[7] Taoying Liu, Jing Liu, Hong Liu, Wei Li, "A performance evaluation of Hive for scientific data management," *Big Data, 2013 IEEE International Conference* on 6-9 Oct. 2013, pp. 39, 46

[8] Anja Gruenheid , Edward Omiecinski , Leo Mark,"Query optimization using column statistics in hive", *Proceedings of the 15th Symposium on International Database Engineering & Applications at Lisboa, Portugal* on September 21-23, 2011,pp. 97,105.

[9] Wenwen Dou, Xiaoyu Wang, Skau D, Ribarsky W, Zhou M.X , "LeadLine: Interactive visual analysis of text data through event identification and exploration," *Visual Analytics Science and Technology (VAST), 2012 IEEE Conference* on 14-19 Oct. 2012, pp.93, 102].