© 2017 IJSRSET | Volume 3 | Issue 6 | Print ISSN: 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Click Through Rate Utilization Using Hadoop Framework on Cloud Environment

Shraddha Sharma, Asst Prof. Abhishek Kumar Dewangan
Department of Computer Science and Engineering, Shri Shankaracharya Technical Campus, Bhilai, Chhattisgarh, India

ABSTRACT

In today's Internet world each one experiences the web straight forwardly or in a roundabout way. Current programming based frameworks gather data about their activity and history in click through file. These files data can be used for analysis. The click through document having timestamps, IP address, month, date, and so forth. In today's web situation click through record examination get to be distinctly vital undertaking for dissecting the client's conduct and for enhancing the web applications and managing an account frameworks, and so forth. Rapid change in the technology, make analysis of these file to predict how system will behave in an uncertain situations. Hadoop and apache spark are well known parallel processing system. HDFS and MapReduce are parallel processing systems. Click through files which are created by the web servers contain information about the activities of the visitors like number of visitors and from which domain they are visiting. Thus analyzing those file are very tedious tasks. This paper analysis wiki dataset over different nodes on AWS cloud for performance evaluation of processing of click through rate dataset over varying number of nodes as well as number of records.

Keywords : Hadoop, HDFS, Hit Count, Click Throgh Document.

I. INTRODUCTION

In today's extending world, everything is going on the web. Sectors like private, public and business has seen a noteworthy improvement in their particular fields. There has been an exponential development in information over the web. We have to dissect these information to give better support of different parts to enhance venture situation. Information is heterogeneous in nature and investigation of such information will give us essential data wherein log documents give an effective arrangement. Log documents are situated in the web server and it contains data about each individual's solicitations, which is put away in a log passage. The fundamental reason for utilizing Hadoop MapReduce is to investigate the datasets successfully.

Click Through documents are produced ordinary which are in the request of terabytes. Click through documents contain colossal measures of valuable data which can helpful to enhance business endeavors and future appraisal. Keeping in mind the end goal to pick up information about the client's exercises, regardless of whether he is buying the item, in the event that he is finding the application cordial to utilize or the issues he is confronting and how it can be settled, we have to investigate click through records. In this manner through click through document investigation, we pick up knowledge into all the previously mentioned inquiries and association of individuals with web applications.

Formula of Click Through rate given by:

$$\text{CTR} = \frac{\text{Clicks}}{\text{Impressions}} * 100$$

CTR – Click through Rate
Clicks – users who click on the link.
Impressions – number of people who viewed link.

Apache Hadoop and its Architecture

The Apache Hadoop programming library is a system that takes into account the distributed processing of extensive information sets crosswise over clusters of PCs utilizing simple programming models. It is designed to scale up from single servers to a large

number of machines, every offering neighborhood computation and storage.

As opposed to depend on equipment to convey high-accessibility, the library itself is intended to recognize and handle failures at the application layer, so conveying an exceedingly accessible administration on top of a cluster of PCs, each of which might be inclined to failure.

MapReduce Architecture

Hadoop MapReduce is a software framework for executing tremendous measure of information i.e. terabyte data sets in parallel environment on large clusters (in a huge number of data nodes) which can be commodity hardware in a fault tolerant manner.

MapReduce jobs splits the input information set into different pieces of files which then are handled by the guide assignments in parallel form. The hadoop framework sorts the output of map phase which are then input to the reduce tasks. Both input and output files are stored on HDFS (Hadoop Distributed File System). The Hadoop framework has a duty of managing and scheduling tasks.

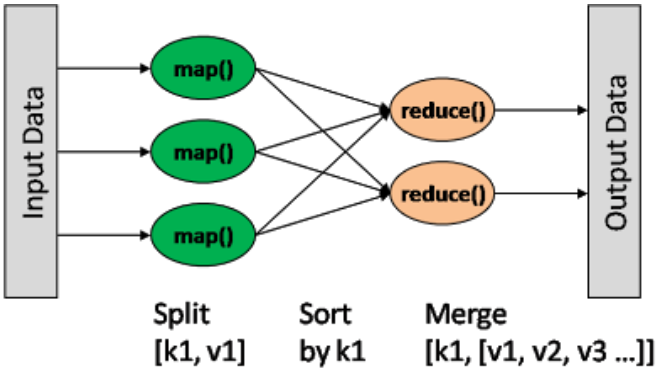

Figure 1. Shows workflow of MapReduce

II. Literature Survey

Lili Shan et al. [1] compare the performance and runtime complexity of our method with Tucker decomposition, canonical decomposition and other popular methods for CTR prediction over real-world advertising datasets. Our experimental results demonstrate that the improved model not only achieves better prediction quality than the others due to considering fully coupled interactions between three entities, user, publisher and advertiser but also can accomplish training and prediction with linear runtime.

Sayalee Narkhede et al. [2] applied this Hadoop MapReduce programming model for analyzing web log files so that we could get hit count of specific web application. This system uses Hadoop file system to store log file and results are evaluated using Map and Reduce function. Experimental results show hit count for each field in log file. Also due to MapReduce runtime parallelization response time is reduced.

Marlina Abdul Latib et al. [3] highlights the characteristics of Big Data as well as Hadoop Framework that has been widely used as Big Data application. Results from the papers reviewed shows that majority researchers applied MapReduce as the main component of Hadoop for analyzing the log files and HDFS as the data storage. Previous researchers have also used other tools and algorithms together with the Hadoop Framework for analyses purposes. The findings of this paper provide a comprehensible review of Hadoop usage performance in analyzing different types of log files and recommend understandable results for end users to use in future work.

Avneesh Tiwari et al., [4] analyses huge datasets need a parallel processing mechanism. Hadoop and apache spark are well known parallel processing system. HDFS and MapReduce are parallel processing systems. In this paper, authors propose a log analysis system which is run over hadoop MapReduce and Apache Spark environment. Both the framework proposes log data in parallel system using all the mechanism in the hadoop and spark cluster and computes result efficiently. Output in the form of some parameters of log such as IP addresses, month, date, time etc so that it can be useful for real time projects, companies, banks etc.

Xuerui Wang et al. [5] develops models and methods to smoothen CTR estimation by taking advantage of the data hierarchy in nature or by clustering and data continuity in time to leverage information from data close to the events of interest. In a contextual advertising system running at Yahoo!, we demonstrate that our methods lead to significantly more accurate estimation of CTRs.

III. Methodology

The proposed workflow is presented in Figure 1. In this paper, our main task is to analyze the hadoop framework performance on the cloud environment. We have used wiki dataset for analysis with various languages. The count of various repeated words are found with matching languages and produces output with the total count with respect to specific language.

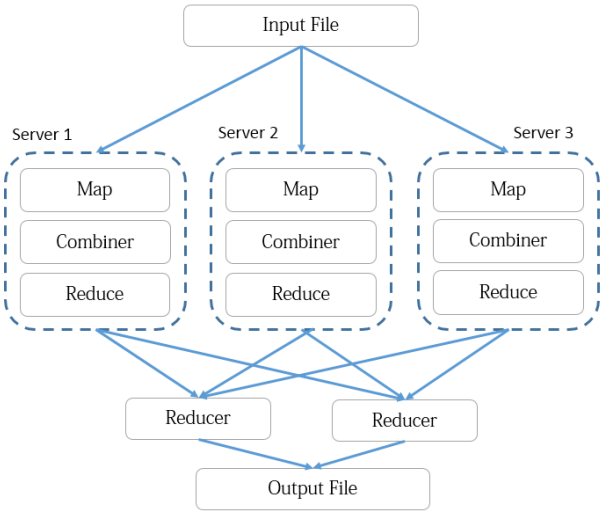

Figure 2. Shows the MapReduce workflow for hit count per language

Working of Mapper

The mapper firstly read the text document line by line with the help of record reader. The mapper reads key value pair. The key is the record length and value is the content of the wiki pages. The mapper simply split using space and tokenize the content and pass to the reducer.

Working of Reducer

The reducer receives the input from the mapper as key, value (iterator) pair, where value is iterator form. The value can be iterated to get the count of all languages. Figure 3 and 4. Shows the input and output file.

```
aa File:Cash_payment_timeline_on_foreclosures.jpg 1 8510
aa File:English-as-Official-Language.png 1 8046
aa Main_Page 7 237627
aa Main_page 2 97609
aa Special:CentralAuth/MSBOT 1 13980
aa Special:Contributions/%D7%93%D7%9F_%D7%90%D7%A4%D7%A8%D7%AA 1
aa Special:Contributions/37.187.147.158 1 6036
```

Figure 3. Shows the snapshot of input file

IV. Result

We have performed experiments on cloud environment. Amazon S3 and EMR are used to store and process file in hadoop distributed systems. We have experimented out dataset with different number of nodes. Figure 4 shows the output file generated by various servers when processed in parallel fashion.

```
aa  27
aa.b    4
aa.d    13
ace 357
aa  2
aa.b    2
ace 30
aa.d    4
ace 9
aa  7
```

Figure 4. Shows the snapshot of output file

The main objective of this paper is to present various file size input and suggest the appropriate number of servers need to run these file size. Table I and II. presents the file size of the input and time required to run those input in the cloud environment.

We have experimented with 2 nodes and 4 nodes. There is only one master node and rest are slave nodes.

TABLE: I. shows the processing time with 2 nodes

File-Name	Number of Records	Time Taken
Wiki-01	2k records	22 sec
Wiki-02	3k records	25 sec
Wiki-03	4k records	26 sec

TABLE: II. Shows the processing time with 4 nodes

File-Name	Number of Records	Time Taken
Wiki-01	2k records	20 sec
Wiki-02	3k records	18 sec
Wiki-03	4k records	17 sec

Figure 5. Shows that as the number of node increases the time taken time for processing the wiki dataset decreases. This analysis will definitely help the business analysis to process the huge amount of dataset over the distributed machines.

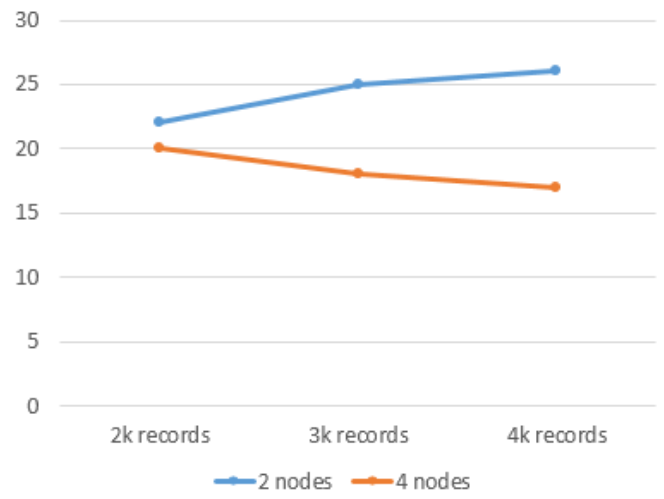

Figure 5. Shows the performance of different nodes on cloud

V. Conclusion

Evaluating CTRs for uncommon occasions are critical however exceptionally difficult. We outlined various strategies to manage the sparsely issue lying in the uncommon occasions. The strategy such as Bayes based empirical formula or Map Reduce Framework based approach, they both are succeeded in process the huge file. We have analyzed our algorithm with varying number of file size and nodes. We have performed experiment over AWS cloud. The server are located at different locations. The outcomes of evaluation is a suggestion for allocating the resources depending upon the size of the file. In out, as the number of nodes increases the processing time decreases hence improving the overall performance of system.

VI. REFERENCES

[1] Lili Shan, Lei Lin, Chengjie Sun, Xiaolong Wang, "Predicting ad click-through rates via feature-based fully coupled interaction tensor factorization", Electronic Commerce Research and Applications archive Volume 16 Issue C, March 2016 , Pages 30-42, ISSN: 1567-4223

[2] Narkhede, S., T. Baraskar, and D. Mukhopadhyay. 2014. Analyzing Web Application Log Files to Find Hit Count through the Utilization of Hadoop MapReduce in Cloud Computing Environment. In 2014 Conference on IT in Business, Industry and Government (CSIBIG),IEEE,MARCH 2015,Page no. 1-7.

[3] Marlina Abdul Latib, Saiful Adli Ismail, Haslina Md Sarkan and Rasimah Che Mohd Yusoff , "ANALYZING LOG IN BIG DATA ENVIRONMENT", ARPN Journal of Engineering and Applied Sciences ,VOL. 10, NO. 23, DECEMBER 2015 ISSN 1819-6608.

[4] Avneesh Tiwari, Rishabh Soni, Dr. Sanjay Agrawal , "WEB LOG MINING USING MAPREDUCE AND APACHE SPARK", International Journal of Engineering Research-Online, Vol.3., Issue.5., 2015 (Sept.-Oct.), ISSN: 2321-7758

[5] Zhipeng Fang, Kun Yue, Jixian Zhang, Dehai Zhang, and Weiyi Liu, "Predicting Click-Through Rates of New Advertisements Based on the Bayesian Network," Mathematical Problems in Engineering, vol. 2014, Article ID 818203, 9 pages, 2014. doi:10.1155/2014/818203