International Journal of Electrical and Computer Engineering (IJECE)
Vol. 10, No. 4, August 2020, pp. 3811~3818
ISSN: 2088-8708, DOI: 10.11591/ijece.v10i4.pp3811-3818

Performance evaluation of Map-reduce jar pig hive and spark with machine learning using big data

Santosh Kumar J.[1], Raghavendra B. K.[2], Raghavendra S.[3], Meenakshi[4]
[1]Department of Computer Science and Engineering, KSSEM, Bangalore, Affiliated to VTU Belagavi, India
[2]Department of Computer Science and Engineering, BGSIT (ACU) Deemed to be University, India
[3]Department of Computer Science and Engineering, Christ Deemed to be University, India
[4]Department of Computer Science and Engineering, Jain Deemed to be University, India

Article Info

Article history:

Received Mar 9, 2019
Revised Feb 1, 2020
Accepted Feb 19, 2020

Keywords:

Cloudxlab
Flink
Hadoop
Hbase
HDFS
Hive
Map-reduce
Pig
Spark

ABSTRACT

Big data is the biggest challenges as we need huge processing power system and good algorithms to make a decision. We need Hadoop environment with pig hive, machine learning and hadoopecosystem components. The data comes from industries. Many devices around us and sensor, and from social media sites. According to McKinsey There will be a shortage of 15000000 big data professionals by the end of 2020. There are lots of technologies to solve the problem of big data Storage and processing. Such technologies are Apache Hadoop, Apache Spark, Apache Kafka, and many more. Here we analyse the processing speed for the 4GB data on cloudx lab with Hadoop mapreduce with varing mappers and reducers and with pig script and Hive querries and spark environment along with machine learning technology and from the results we can say that machine learning with Hadoop will enhance the processing performance along with with spark, and also we can say that spark is better than Hadoop mapreduce pig and hive, spark with hive and machine learning will be the best performance enhanced compared with pig and hive, Hadoop mapreduce jar.

Corresponding Author:

Santosh Kumar J.,
Department of Computer Science and Engineering,
KSSEM Bengaluru VTU University,
Mallasandra, Kanakapura Road, Bangalore-560109, India.
Contact: +919035636616
Email: santosh.kumar.j@kssem.edu.in

1. INTRODUCTION

Big data refers to data sets whose size is beyond the ability of typical database management tools to capture, store, manage, and analyze. Cloud computing and big data, two disruptive trends at present, pose significant influence on current IT industry and research communities. Cloud computing provides massive computation power and storage capacity which enable users to deploy applications without infrastructure investment. Integrated with cloud computing, data sets have become so large and complex that it is a considerable challenge for traditional data processing tools to handle the analysis pipeline of these data. Generally, such data sets are often from various sources and of different variety such as unstructured social media content and semi-structured medical records and business transactions are of large volume with fast data [1].

The Map Reduce framework has been widely adopted by a large number of companies and organizations to process huge volume of data sets. Unlike the traditional Map Reduce framework, the one incorporated with cloud computing becomes more flexible, salable and cost-effective. A typical example is the Amazon Elastic Map Reduce service. Users can invoke amazon EMR to conduct their Map-reduce computations based on the powerful infrastructure offered by Amazon. Web Services and are charged in

proportion to the usage of the services. In this way, it is economical and convenient for companies and organizations to capture, store, organize, share and analyze big data to gain competitive advantages. Map Reduce is currently a major big data processing paradigm. The authors discussed about existing performance models for Map Reduce only comply with specific workloads that process a small fraction of the entire data set, thus failing to assess the capabilities of the Map Reduce paradigm under heavy workloads that process exponentially increasing data volumes. The authors discussed about building and analyze a scalable and dynamic big data processing system, including storage, execution engine, and query language. The authors mainly concentrated in the design and implementation of a resource management system, design and implementation of a bench marking tool for the Map Reduce processing system and the evaluation and modeling of Map Reduce using workloads with very large data sets [2]

Spark is the 100 times faster framework than Map Reduce and hdfs in storage and processing it is also frame work like any other java framework which built on top of OS to utilize memory efficiently and the other devices of CPU efficiently particularly designed framework for big data processing. Spark has many advantages and disadvantages efficient utilizations of memory management is one of the disadvantage of spark whereas processing big data is advantages compared with map reduce framework and HDFS of Hadoop.

Flink is also a frame work for all components of Hadoop eco-system. Flink is the frame work for Streaming data, flinklatency is very less to process big data compared with Spark Flink has many advantages, it processes the data without latency like speed of light, and Memory exception problem is also solved by Flink. Flink also interact with many devices of which have different storage system to process the data, and it also optimizes the program before execution.

Big data Processing Technology like Hadoop mapreduce, flink and spark along with caching data processing engine and scheduler as shown in Figure 1. Data Processing technique like data understanding data peploration and data modeling are as shown in Figure 2. Big data Ecosystem components like pig hive spark ambari zookeeper ml lib Habase and many as shown in Figure 3.

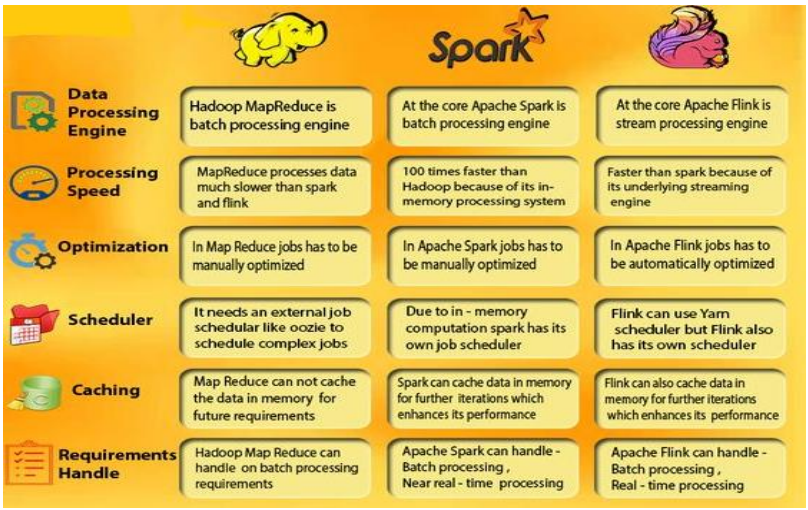

Figure 1. Big data processing technology comparision

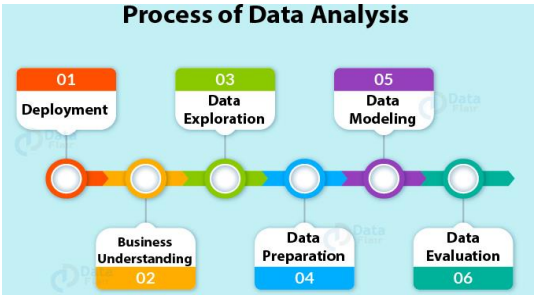

Figure 2. Data analysis processing steps

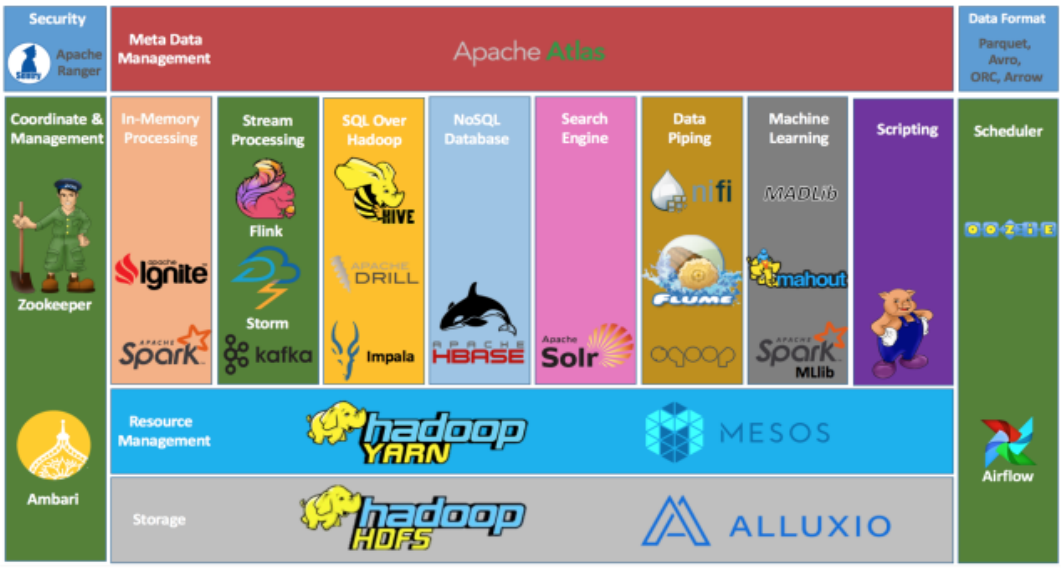

Figure 3. Eco system components of big data

2. LITERATURE REVIEW

Many author of the paper said about Apache Hadoop that it is a framework for processing large distributed data set across cluster of computers and said about scaling the cluster. Due to use of sensors across all devices and network tools of the organizations generating big data, all wanted to store and analyze without investing much cost on managing and service issue of the storage and processing want to deploy everything on cloud so that cloud management organizations will take care of it, these companies can utilize the data for analysis and extract useful knowledge out of it. Map Reduce is the framework which allows large data to be stored across all devices and processed by devices map functions will distribute the data and store across the devices where a reduce will process the query of the client it works on bases of the key value pair. Each line will be treated as key and value that is first word is the key and rest all will be value whenever client request to process the large data first client will approach the name node name node will respond to client with available free nodes after that mapper functions by client will write data to respective data nodes, and whenever client want to process the data it request to name node job tracker then job tracker will communicate to name node to get data information storage then it will assign jobs to task tracker to process the job by name nodes will process the task by their available data then one of the node will aggregate the result and give the result to client [3].

Hadoop's optimization framework for Map Reduce clusters the author of the paper states most widely used frameworks for developing Map Reduce based applications is Apache Hadoop. But developers find number of challenges in the Hadoop framework, which causes problem to management of the resources in the Map Reduce cluster that will optimize the performance of Map Reduce applications running on it. The constraints in the resource allocation process in the Map Reduce programming model for large-scale data processing for speed up performance. The novel technique called Dynamic approach for performing speed up of the available resources. It contains the two major operations; they are slot utilization optimization and utilization efficiency optimization. The Dynamic technique has the three slot allocation techniques they are dynamic hadoop slot allocation speculative execution performance balancing and Slot Pre-scheduling. It achieves a performance speedup by a factor of over the recently proposed cost-based optimization approach. In addition, performance benefit increases with input data set size [4].

Performance Evaluation of Hadoop and Oracle Platform for Distributed Parallel Processing in big data Environments The authors discussed about the Reduce data center implementation cost using commodity hardware to provide high performance Computing. Distributed processing of large data sets across clusters of computers using distributed and parallel computing architecture. And also the authors do the Performance comparison of distributed parallel computing system and traditional single computing system towards an optimized big data processing system author of the paper stated that the authors discussed about resource management system for Map Reduce based processing system for deploying and resizing Map Reduce clusters Bench marking tool for the Map Reduce processing system evaluation and modeling of Map Reduce using workloads with very large data sets and to optimize the Map Reduce system to efficiently process terabytes of data. Overview on performance testing approach in big data the author stated that many

organizations are facing challenges in facing test strategies for structured and unstructured data validation, setting up optimal test environment, working with non relational database and performing non functional testing. These challenges cause poor quality of data in production, delay in implementation and increase in cost. Map Reduce provides a parallel and scalable programming model for data-intensive business and scientific applications. To obtain the actual performance of big data applications, such as response time, maximum online user data capacity size, and a certain maximum processing capacity [5]. The paper authors discussed big data and computing cloud management appliances and the processing problems of big data, with reference to computing cloud, database of cloud, cloud architecture, Map Reduce optimization techniques [6]. The authors discussed the Resource management Mappers and Reduce-based applications processing to deploy and resizing Map Reduce Bench marking applications and tool are used for the Map Reduce processing to extent the Map Reduce enactment using workloads with big data and to optimize the Map Reduce to process terabytes of data proficiently and Cost Optimizations for Workflows in the Cloud [7, 8]. The authors discussed about software to expand the scalability of data analytics, Challenges Availability, partitioning, virtualization and scalability, distribution, and elasticity and performance bottlenecks for managing big data [9]. The authors said about Benchmarking a several of high-performance computing (HPC) architectures for data, name node and data node architectures with large memory and bandwidth are better suited for big data analytics on HPC h/w and Budget-Driven Scheduling Algorithms for Batches of MapReduce Jobs in Heterogeneous Clouds [10, 11]. Map Reduce provides a parallel and scalable programming model for data-intensive business and scientific applications. To obtain the actual performance of big data applications, such as response time, maximum online user data capacity size, and a certain maximum processing capacity [12]. On the other paper author have discussed about the parallel processing techniques [13]. Other author of the paper discussed about performance issue with Cloud and big data [14]. The author said about tesing techniques and performance enhancement parameters [15] and the aother authors discussed about multicore architecture of Hadoop performance [16]. The author discuused about the Machine learning techniques with Hadoop may enhances the performance [17].The author of the paper said about Hadoop self tuning mapper and reducer with mland clustersof architectur and optimization of big data performance parameters [18, 19].The author discussed the performance with oracle and Hadoop and said Hadoop enhances the performance [20]. The authors discussed Map-reduce execution time Big.txt input file. With cloudxlab Hadoop big data frame work [21]. The authors discussed Map-reduce execution time Ramayana text input file. With cloudxlab Hadoop big data frame work [22]. The author discussed about the AWS Costbased Optimization of Map-Reduce Programs may enhance the performance [23]. The author said about efficient utilization of mapper and reducer may enhance the performance [24]. The author discussed about Resource-aware Adaptive Scheduling for MapReduce Clusters [25]. The author discussed about performance of Pig hive and Hadoop jar file [26].

3. RESULTS AND DISCUSSION

Figure 4 is the Map Reduce architectural framework for word count program where hugeinput file is split as blocks of pages and each pages split as lines and each lines spit as words by spaces to get number of words then all words are shuffled with all the data nodes mappers to count occurrence of each words in each data nodes finally using reduces combines the results achieved by each data node. Running Character Count Job in Cloudxlab hadoop jar /usr/hdp/2.3.4.0-3485/hadoop-mapreduce/hadoop-streaming.jar-input/data/mr/ wordcount/input -output letter_count -mapper mapper.py -file mapper.py -reducer reducer.py file reducer.py. The Table 1 shows the out of the character count job, which reads the input file and calculate the number of occurrences of the character and store the output in output file. Figures 4-7 shows the execution time of word count program of pig script and Hive Query. First, we create a table called doc then will load a input file after that word count query program execution which shows a time of 14 Sec to exec. Total of 20 Sec to execute a word count program for input file (14sec+ 6sec= 20 Sec). Total of 36 sec + 16 sec = 52 sec of time to execute the word count program for input file. Table 1 shows the characters and its count on mapreduce Hadoop after execution.

Mapreduce framework for the word count as shown in Figure 4, huge input data is divided and given to mapper based on key value pair for the data. Then the suffle action later reducer will be used for combining the results of mapper. The word count program is given for the execution with Spark Hive and machine learning query and Execution time of 6 seconds as shown in Figure 5. The word count program is given for the execution with Spark Hive query and Execution time of 14 seconds as shown in Figure 6. The word count program is given for the execution with Spark Hive query and Execution time of 16 seconds as shown in Figure 7. The word count program is given for the execution with PIG query and Execution time of 36 seconds as shown in Figure 8.

Table 1. Character Count output

Character and its Count	Character and its Count
a. 08096	n. 369018
b. 73168	o. 386867
c. 144974	p. 98913
d. 215706	q. 4571
e. 633821	r. 309558
f. 120875	s. 334901
g. 96916	t. 460748
h. 294683	u. 138732
i. 365641	v. 52378
j. 6436	w. 100831
k. 32798	x. 9810
l. 198648	y. 90481
m. 127063	z. 3796

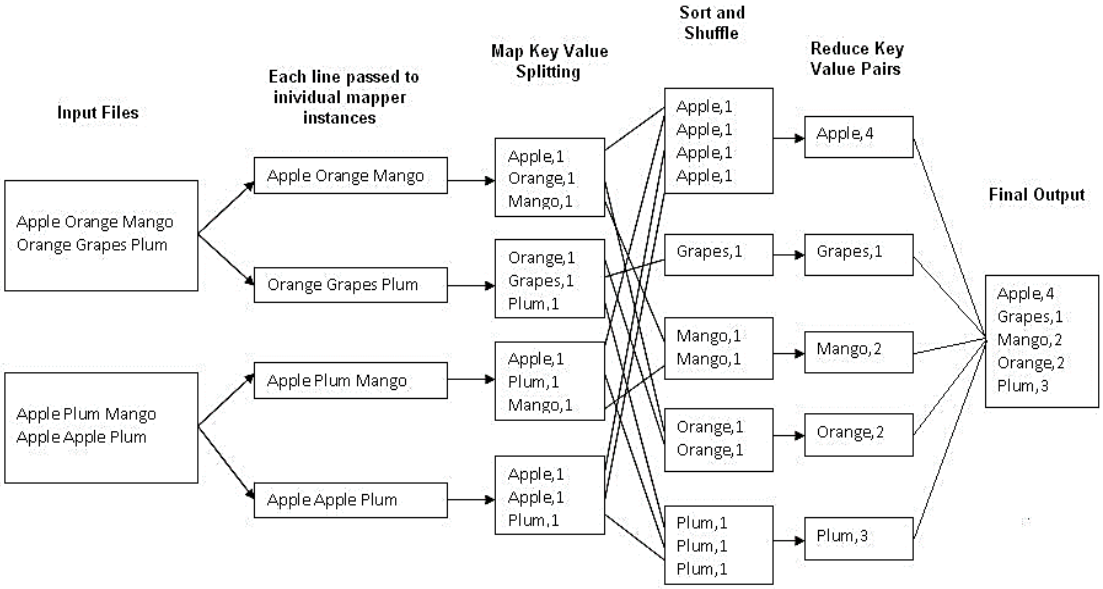

Figure 4. Map Reduce framework for word count

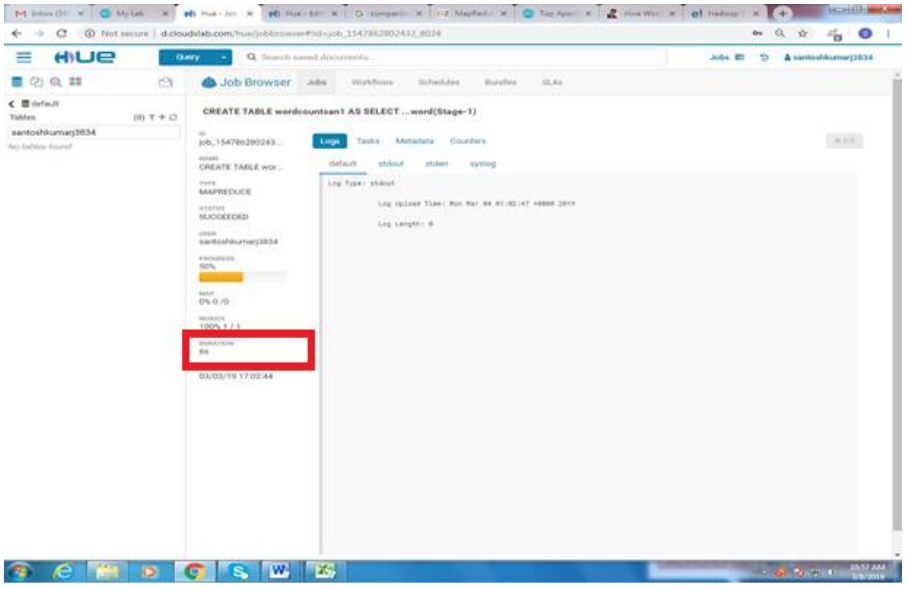

Figure 5. Hive Query execution time 6 sec for input file

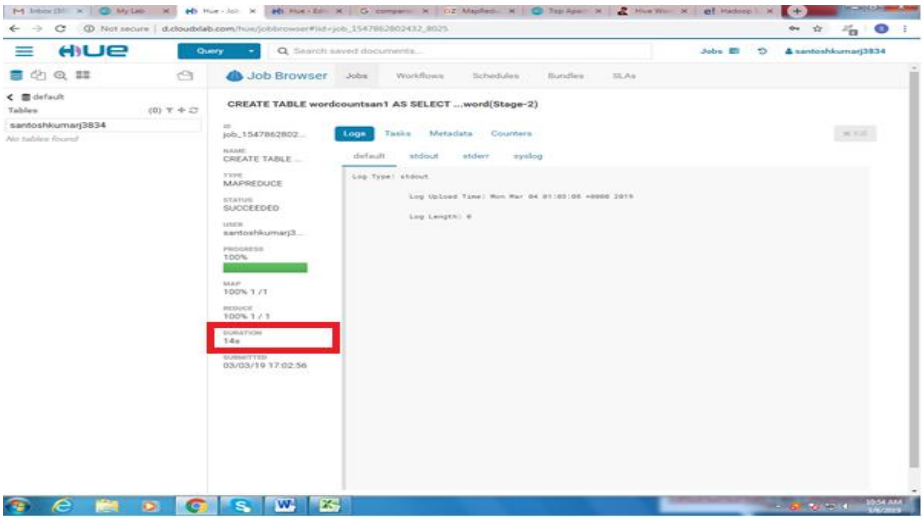

Figure 6. Hive Query execution time 14 Sec for input file

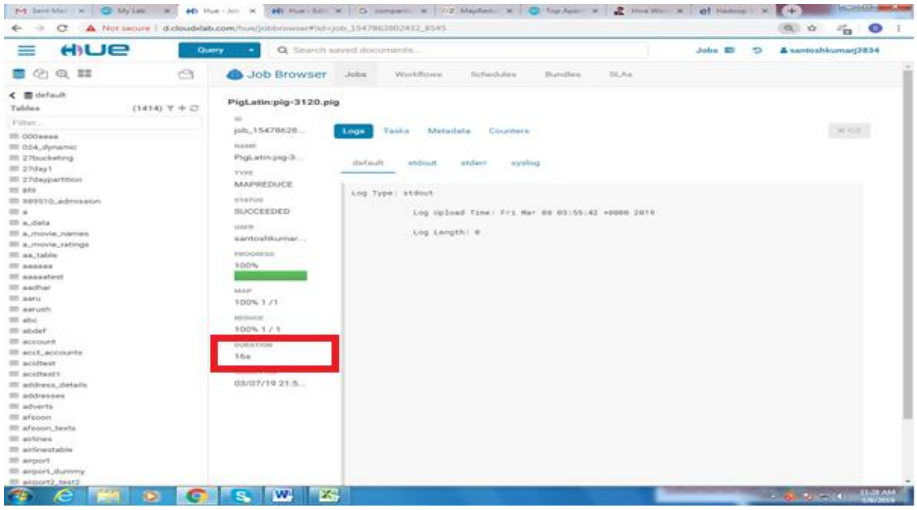

Figure 7. Word count program execution time 16 Sec for input file

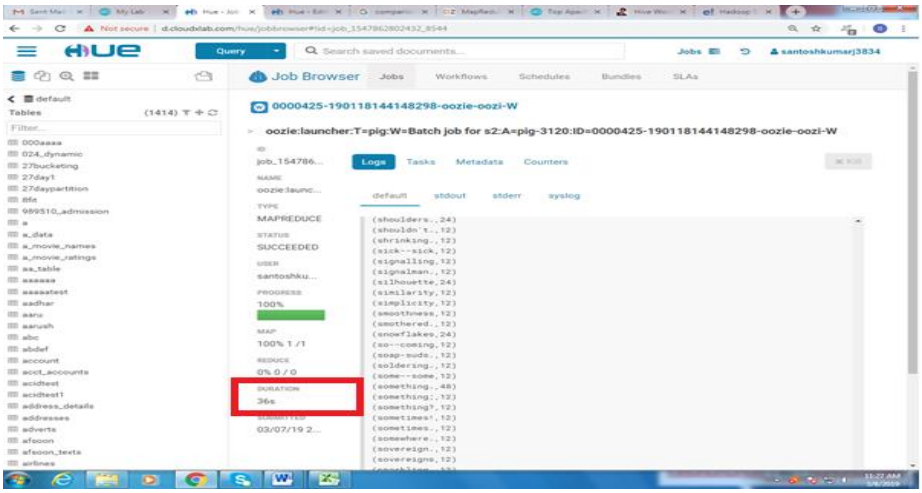

Figure 8. Word count program execution time 36 Sec for the input file

4. CONCLUSION

Hadoopissoftware framework for variety, volume and velocity of data processing, companies like google yahoo and Amazon have their own framework for processing the big data also they provide cloud based big data eco-system infrastructure to store (using HDFS) and process (using map-Reduce) big data, from above results we say that Hive Query execution time is 20 Seconds, whereas pig script execution time is 52 Seconds for the same input file without machine learning and with machine learning its enhanced to 16 seconds with combination of ml and spark with hive also, we can say that word count program for given input file Hive is better than Pig, Hive enhances the execution time, from above results we can we may state that machine learning, spark with hive gives enhanced performance than hadoop mapreduce and pig spark and flink.

ACKNOWLEDGEMENTS

I would like express my deep gratitude to the Principal, HoD and Staff of Computer Science and Engineering department of KSSEM, Bangalore for supporting me in doing this research work.

REFERENCES

[1] Md. Armanur Rahman, J. Hossen, "A Survey of Machine Learning Techniques for Self-tuning Hadoop Performance," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 3, pp. 1854-1862, 2018.
[2] Aman Lodha, "Hadoop's Optimization Framework for Map Reduce Clusters," *Imperial Journal of Interdisciplinary Research (IJIR)*, vol 3, no 4, pp. 1648-1650, 2017.
[3] Dan Wang, Jiangchuan Liu, "Optimizing Big Data Processing Performance in the Public Cloud: Opportunities and Approaches," *IEEE Network*, September/October 2015.
[4] A. K. M. Mahbubul Hossen, A. B. M. Moniruzzaman et. al., "Performance Evaluation of Hadoop and Oracle Platform for Distributed Parallel Processing in Big Data Environments," International *Journal of Database Theory and Application*, vol. 8, no. 5, pp.15-26, 2015.
[5] Changqing Ji, Yu Li, Wenming Qiu et.al., "Big Data Processing in Cloud Computing environments," *International Symposium on Pervasive Systems, Algorithms and Networks*, 2012.
[6] Aman Lodha, "Hadoop's Optimization Framework for Map Reduce Clusters" *Imperial Journal of Interdisciplinary Research (IJIR)*, vol-3, no 4, 2017
[7] Dan Wang, Jiangchuan Liu, "Optimizing Big Data Processing Performance in the Public Cloud: Opportunities and Approaches" *IEEE Network*, September/October 2015.
[8] C. Zhou, B.S. He, "Transformation-based Monetary Cost Optimizations for Workflows in the Cloud," *IEEE Transaction on Cloud Computing*, 2014.
[9] A. K. M. MahbubulHossen1, A. B. M. Moniruzzaman et. al. "Performance Evaluation of Hadoop and Oracle Platform for Distributed Parallel Processing in Big Data Environments," *International Journal of Database Theory and Application*, vol. 8, no. 5, pp.15-26, 2015.
[10] Changqing Ji, Yu Li, Wenming Qiu et.al. "Big Data Processing in Cloud Computing environments," *International Symposium on Pervasive Systems, Algorithms and Networks*, 2012
[11] Y. Wang, W. Shi, "Budget-Driven Scheduling Algorithms for Batches of MapReduce Jobs in Heterogeneous Clouds," *IEEE Transaction on Cloud Computing*, 2014
[12] Bogdan Ghiţet. al. "Towards an Optimized Big Data Processing System" 13th IEEE/ACM International Symposium on Cluster, Cloud, and Grid Computing, 2013
[13] Kyong-Ha Lee et. al. "Parallel Data Processing with Map Reduce: A Survey," *SIGMOD Record*, vol. 40, no. 4, December 2011.
[14] Jaliya Ekanayake and Geoffrey Fox "High Performance Parallel Computing with Clouds and Cloud Technologies" *International Conference on Cloud Computing*, 2009.
[15] Ashlesha S. Nagdive et al, "Overview on Performance Testing Approach in Big Data," *International Journal of Advanced Research in Computer Science*, vol. 5, no. 8, Nov–Dec, pp. 165-169, 2014.
[16] Y. Zhang, "Optimized runtime systems for MapReduce applications in multi-core clusters," Thesis, Rice University, Texas. 2014. [Online]. Available: https://www.semanticscholar.org/paper/Optimized-Runtime-Systems-for-MapReduce-in-Clusters-Zhang/10fa14d4c7846bf9b35e8507a8dcdcb7ee79a672
[17] Md. Armanur Rahman, and J. Hossen, "A Survey of Machine Learning Techniques for Self-tuning Hadoop Performance," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 3, pp. 1854-1862, June 2018.
[18] Aman Lodha, "Hadoop's Optimization Framework for Map Reduce Clusters," *Imperial Journal of Interdisciplinary Research (IJIR)*, vol-3, no.-4, pp. 1648-1650, 2017.
[19] Dan Wang, Jiangchuan Liu , "Optimizing Big Data Processing Performance in the Public Cloud: Opportunities and Approaches*" IEEE Network*, September/October 2015.
[20] A. K. M. Mahbubul Hossen, A. B. M. Moniruzzaman et. al., "Performance Evaluation of Hadoop and Oracle Platform for Distributed Parallel Processing in Big Data Environments," *International Journal of Database Theory and Application*, vol. 8, no. 5, pp.15-26, 2015.

[21] J. Santosh Kumar, S. Raghavendra, B. K. Raghavendra et.al., "Big data Performance Evalution of Map-Reduce Pig and Hive," *International Journal of Engineering and Advanced Technology (IJEAT)*, vol-8, no.-6, August 2019.
[22] J. Santosh Kumar, S. Raghavendra, B. K. Raghavendra et.al., "Big data Processing Comparison using Pig and Hive," *International Journal of Computer Science and Engineering (IJCSE)*, vol. 7, no. 3, March 2019.
[23] H. Herodotou and S. Babu, Profiling, "What-if Analysis, and Costbased Optimization of MapReduce Programs," In *Proc. of the VLDB Endowment*, vol. 4, no. 11, 2011.
[24] Z. H. Guo, G. Fox, M. Zhou, Y. Ruan, "Improving Resource Utilization in MapReduce," In *IEEE Cluster'12*, pp. 402-410, 2012.
[25] J. Polo, C. Castillo, D. Carrera, et al., "Resource-aware Adaptive Scheduling for MapReduce Clusters," In *Middleware'11*, pp. 187-207, 2011.
[26] J. Santosh Kumar, S. Raghavendra, B. K. Raghavendra et. al., "Big data Performance evaluation of mapreduce pig hive," *International Journal of Engineering and Advanced Technology (IJEAT)*, vol. 8 no. 6, Aug. 2019.

BIOGRAPHIES OF AUTHORS

Santosh Kumar J.is currently working as Associate Professor in the Department of Computer Science and Engineering at K.S. School of Engineering and Management, Bangalore. Affiliated to VTU Belagavi, He is pursuing Ph.D. in VTU, Belgaum, India. He has 10 years of teaching and 3 years of industry experience. He isInterested in Big data streaming analysis. His reserach topics includesBig data with machine learning.

Dr. Raghavendra B.K. Pursued P. hd From Dr. MGR Educational & Research Institute, Chennai and Masters from PESCE, Mandya bengaluru university and Bachelors from GCE, Ramanagara Bengaluru University Karnataka He published nearly 15 reputed journals and His Area of interest is Data mining and Big data He currently working in BGSIT B G Nagar (ACU) Mandya as Professor and Head department computer science and engineering.

Dr. Raghavendra S. is currently working as Associate Professor in the Department of Computer Science and Engineering at CHRIST DEEMED TO BE UNIVERSITY, Bangalore. He completed his Ph.D. degree in Computer Science and Engineering from VTU, Belgaum, India in 2017 and has 14 years of teaching experience. His interests include Data Mining and Big data.

Meenakshi is currently working as Assistant Professor in the Department of Computer Science and Engineering specialization at Jain Deemed to be university Bangalore. She completed masters from VTU Belagavi and has 1 year of teaching experience. she is Interested in Big data streaming analysis.