Performance Comparison of Hive, Pig & Map Reduce over Variety of Big Data

Yojna Arora, Dinesh Goyal

***Abstract*: Big Data refers to that huge amount of data which cannot be analyzed by using traditional analytics methods. With the increase of web content at a rapid rate, only analyzing data is not enough rather managing it with that great pace and efficiency is needed. A new framework Hadoop was implemented in order to perform parallel distributed computing. Hadoop is supported by various frameworks. In this paper, a performance comparison of Pig, Hive and Map Reduce over Big Data is analyzed.**

***Index Terms*: Pig, Hive, Map Reduce, Hadoop, Big Data**

I. INTRODUCTION

Hadoop is open source software which helps to process large data sets across cluster of computers using simple programming paradigm. Hadoop was driven by the concept of Google File System. Hadoop was inspired by a paper published by Google which mentioned the approach of dealing with Big Data. It relies on distributed computing performed over low cost servers. Storage, Management and Processing are the three major challenges of Big Data which are successfully handled by Hadoop [1].

Features of Hadoop

Fault Tolerance: The capability of Hadoop to support multiple nodes running at same time provides fault tolerance. In case of any node failure, the system assigns the task to other node without any interruption or delay in the job. It is achieved by Data replication and redundancy.

Scaling: The distributed file system behavior of Hadoop helps it to add or delete nodes as required

Move Compute: Hadoop deals with Big Data so the overhead needed to move the data is much more than the computation itself. Keeping this in mind the computational queries are moved to the place where data resides. They are executed in distributed manner.

A. APACHE hIVE

HIVE has a three layered architecture. The User Interface Layer, the Processing layer and the Database layer. The first layer provides an Interface to the End user to write queries for data analysis. The middle layer provides the functional support to the architecture.

Revised Manuscript Received on 30 January 2019.
* Correspondence Author

Dr. Dinesh Goyal, Professor, Department of Computer Science and Engineering, Poornima Institute of Engineering and Technology, Jaipur (Rajasthan), India.

Dr. Yojna Arora, Department of Computer Science, College of Computer Science and Engineering, Amity University, Haryana, India.

Meta Store contains the information regarding schema of the database tables. All relevant information required for querying like column details, data types etc.

HiveQL supports querying on schema information on Meta Store. It is used for executing query on Map Reduce Platform. It is a replacement for Map Reduce. The execution engine works as a conjunction between HIVEQL and Map Reduce. It generates corresponding Map Reduce result for the issued query. The last layer HDFS works is backend storage to store the data which needs to e analyzed.

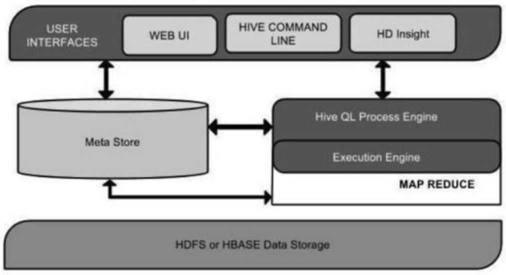


Fig 1. Hive Architecture

Many consider Hive to be much more effective for processing structured data than unstructured data, for which Pig is considered advantageous.

In [2], Hive framework was introduced. Hive is a data warehousing framework build on top of Hadoop. It also explains the operation of Hive QL which executes SQL like queries and also provides plugins for writing custom map reduce. HIVE system architecture is given along with the execution of various SQL like queries on data set.

B. APACHE PIG

Pig is a procedural language for developing parallel processing applications for large datasets in Hadoop environment. Pig is an alternative to Java programming for Map Reduce, and automatically generates Map Reduce functions. Pig includes Pig Latin, which is a scripting language. Pig translates Pig Latin scripts into MapReduce, which can then run on YARN and process data in the HDFS cluster. Pig is popular because it automates some of the complexity in MapReduce development. Pig is commonly used for complex use cases that require multiple data operations. It is more of a processing language than a query language. Pig helps develop applications that aggregate and sort data and supports multiple inputs and outputs. It is highly customizable, because users can write their own functions using their preferred scripting language. Ruby, Python and even Java are all supported [3].

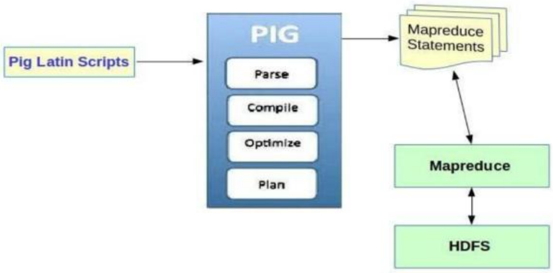


Fig 2, Pig Architecture

The language supported by Pig, Pig Latin contains various in built operators which perform the functions to load, store etc. The above Figure explains the architecture of Pig. Initially, the scripts go to the parser which checks it for any syntax or semantics errors by generating parse tree from the tokens. The parse tree thus generated, goes as an input to the Compilation phase in which the compiler checks all the meanings and verify the user defined functions for any further errors. Later on, in the optimize phase, an optimization procedure is executed hitch eliminates all the meaningless code. Some of the optimization techniques are dead code elimination, loop optimization etc. Lastly, planning is done in the plan phase. The compiled code is then sending to Map Reduce and the job is executed.

C. Map Reduce

In Map Reduce in [4], the first step is the map job which takes a set of data and converts it into another set of data, where individual elements are broken down into smaller tasks (key value pair). The reduce jobs then takes the output from a map as input and combine these into a smaller set of results. The map function can run independently on each key value pair, allowing parallelism. Reduce function can also implement on each Intermediate key. It is a linearly scalable programming model. Mathematically it can be explained as a one to one mapping of Key Value pair. The Map reduce process is shown in Fig 3 below

Map: (k1, v1) → [(k2, v2)

Reduce: (k2, [v2]) → [(k1, v1)]

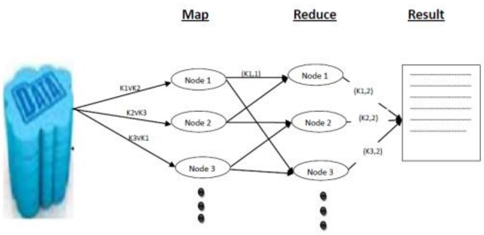


Fig 3 Map Reduce Process

Map Reduce guarantee the delivery of data to reducer in sorted order. The result is then send to reducer in sorted order itself. HDFS stores the result. Map Reduce can also ensure parallelism. If a node is working slower than the other nodes available in cluster, then other nodes can share the jobs providing parallelism and reducing overheads.

Big Data Analytics using other components of Hadoop is explained in detail in [5] & [6].

II. METHODOLOGY

Big Data Analytics is termed as analyzing the data in order to fetch valuable information from it [7]. The nature of data plays an important role while selecting the technology to be used for analysis. In this work, six very dive rse datasets and taken into consideration. Two data sets each of the three formats structured, semi structured and unstructured as used.

Structured Data

- A Colon separated file which contains the details about Movie Ratings
- An excel spreadsheet which contains data corresponding to an educational institution

Semi Structured Data

- An XML file which contains the details of Books
- A JSON file containing Web Page Information

Unstructured Data

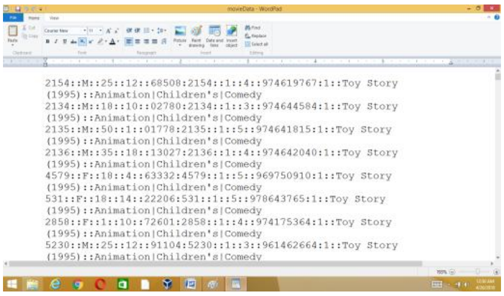
- A plain text file containing a paragraph of text
- A CSV file containing records of an employee

A. Structured Data Description

1. Structured Data Set 1

The dataset which is used as structured data format contains the information about Movies and its ratings given by the user. It is analyzed on all the three platforms to give same result but using different methodology.

Query: To list movie names by its rating in descending order



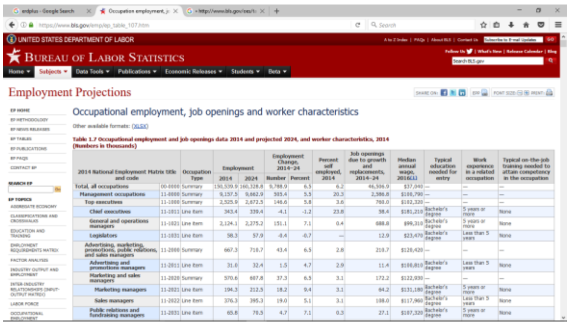
```
2154::M::25::12::68508:2154::1::4::974619767:1::Toy Story
(1995)::Animation|Children's|Comedy
2134::M::18::10::02780:2134::1::3::974644584:1::Toy Story
(1995)::Animation|Children's|Comedy
2135::M::50::1::01778:2135::1::5::974641815:1::Toy Story
(1995)::Animation|Children's|Comedy
2136::M::35::18::13027:2136::1::4::974642040:1::Toy Story
(1995)::Animation|Children's|Comedy
4579::F::18::4::63332:4579::1::5::969750910:1::Toy Story
(1995)::Animation|Children's|Comedy
531::F::18::14::22206:531::1::5::978643765:1::Toy Story
(1995)::Animation|Children's|Comedy
2858::F::1::10::72601:2858::1::4::974175364:1::Toy Story
(1995)::Animation|Children's|Comedy
5230::M::25::12::91104:5230::1::3::961462664:1::Toy Story
(1995)::Animation|Children's|Comedy
```

Fig 4. Structurd Dataset 1

2. Structured Data Set 2

The second dataset which is taken as an example for structured data contains details about people living in an area their occupational employment, job opening and worker characteristics.

Query 1: Analyze the entire data set and arrive at all designations where the annual salary is above $ 50000.



Employment Projections

Occupational employment, job openings and worker characteristics

Other available formats: (XLSX)

Table 1.7 Occupational employment and job openings data 2014 and projected 2024, and worker characteristics, 2014 (Numbers in thousands)

2014 National Employment Matrix title and code		Occupation Type	Employment		Employment Change, 2014–24		Percent self employed, 2014	Job openings due to growth and replacements, 2014–24	Median annual wage, 2016[1]	Typical education needed for entry	Work experience in a related occupation	Typical on-the-job training needed to attain competency in the occupation
			2014	2024	Number	Percent						
Total, all occupations	00-0000	Summary	150,539.9	160,328.8	9,788.9	6.5	6.2	46,506.9	$37,040	—	—	—
Management occupations	11-0000	Summary	9,157.5	9,662.9	505.4	5.5	20.3	2,586.8	$100,790	—	—	—
Top executives	11-1000	Summary	2,525.9	2,672.5	146.6	5.8	3.6	760.0	$102,320	—	—	—
Chief executives	11-1011	Line item	343.4	339.4	-4.1	-1.2	23.8	58.4	$181,210	Bachelor's degree	5 years or more	None
General and operations managers	11-1021	Line item	2,124.1	2,275.2	151.1	7.1	0.4	688.8	$99,310	Bachelor's degree	5 years or more	None
Legislators	11-1031	Line item	58.3	57.9	-0.4	-0.7	—	12.9	$23,470	Bachelor's degree	Less than 5 years	None
Advertising, marketing, promotions, public relations, and sales managers	11-2000	Summary	667.3	710.7	43.4	6.5	2.8	210.7	$120,420	—	—	—
Advertising and promotions managers	11-2011	Line item	31.0	32.4	1.5	4.7	2.9	11.4	$100,810	Bachelor's degree	Less than 5 years	None
Marketing and sales managers	11-2020	Summary	570.6	607.8	37.3	6.5	3.1	172.2	$122,930	—	—	—
Marketing managers	11-2021	Line item	194.3	212.5	18.2	9.4	3.1	64.2	$131,180	Bachelor's degree	5 years or more	None
Sales managers	11-2022	Line item	376.3	395.3	19.0	5.1	3.1	108.0	$117,960	Bachelor's degree	Less than 5 years	None
Public relations and fundraising managers	11-2031	Line item	65.8	70.5	4.7	7.1	0.3	27.1	$107,320	Bachelor's degree	5 years or more	None

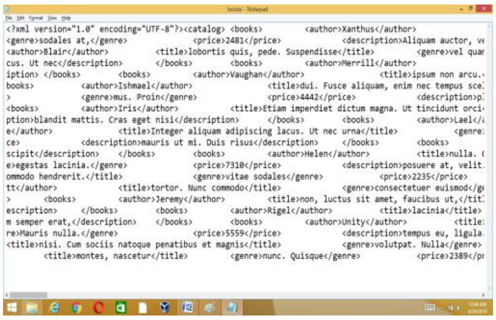
Fig 5 Structured Dataset 2

B. Semi Structured Data Description

1. Semi Structured Data Set 1

In this work, two sets of semi structured data are taken into consideration. One is an XML file which contains the information about books. The dataset which is taken as Semi Structured data is an XML file. It contains the information about Books. The same file is analyzed over all the three platforms Pig, HIVE and Map Reduce

Query: To find the total price of all books written by an author



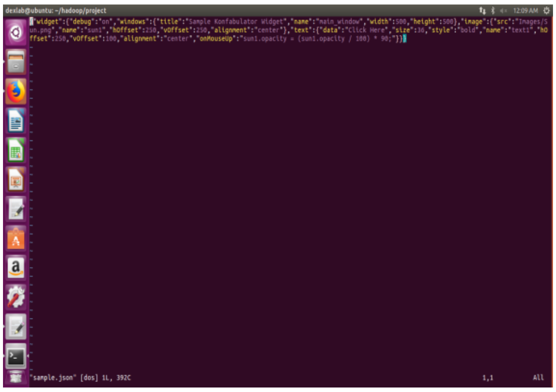
```
<?xml version="1.0" encoding="UTF-8"?><catalog> <books>        <author>Xanthus</author>
<genre>sodales at,</genre>            <price>2481</price>            <description>Aliquam auctor, ve
<author>Blair</author>        <title>lobortis quis, pede. Suspendisse</title>        <genre>vel quar
cus. Ut nec</description>     </books>        <books>        <author>Merrill</author>
iption> </books>        <books>        <author>Vaughan</author>                <title>ipsum non arcu.
books>          <author>Ishmael</author>                <title>dui. Fusce aliquam, enim nec tempus sce
>               <genre>mus. Proin</genre>               <price>4442</price>            <description>p
<books>         <author>Iris</author>          <title>Etiam imperdiet dictum magna. Ut tincidunt orci
ption>blandit mattis. Cras eget nisi</description>      </books>        <books>        <author>Lael</
e</author>              <title>Integer aliquam adipiscing lacus. Ut nec urna</title>          <genre
ce>             <description>mauris ut mi. Duis risus</description>     </books>        <books>
scipit</description>    </books>        <books>         <author>Helen</author>         <title>nulla.
e>egestas lacinia.</genre>            <price>7310</price>            <description>posuere at, velit.
ommodo hendrerit.</title>             <genre>vitae sodales</genre>           <price>2235</price>
tt</author>             <title>tortor. Nunc commodo</title>            <genre>consectetuer euismod</g
>       <books>         <author>Jeremy</author>        <title>non, luctus sit amet, faucibus ut,</tit
escription>     </books>        <books>        <author>Rigel</author>          <title>lacinia</title>
m semper erat,</description>    </books>        <books>        <author>Unity</author>         <title
re>Mauris nulla.</genre>              <price>5559</price>            <description>tempus eu, ligula.
<title>nisi. Cum sociis natoque penatibus et magnis</title>          <genre>volutpat. Nulla</genre>
       <title>montes, nascetur</title>        <genre>nunc. Quisque</genre>          <price>2389</pr
```

Fig 6. Semi Structured Dataset 1

2. Semi Structured Dataset 2

The JSON file is neither organized nor unorganized so it is categorized as semi structured data. The second dataset which is taken for the study is a JSON file.

Query: To count the total number of records



```
{"widget":{"debug":"on","windows":{"title":"Sample Konfabulator Widget","name":"main_window","width":500,"height":500},"image":{"src":"Images/S
un.png","name":"sun1","hOffset":250,"vOffset":250,"alignment":"center"},"text":{"data":"Click Here","size":36,"style":"bold","name":"text1","hO
ffset":250,"vOffset":100,"alignment":"center","onMouseUp":"sun1.opacity = (sun1.opacity / 100) * 90;"}}}
"sample.json" [dos] 1L, 392C                                                          1,1           All
```

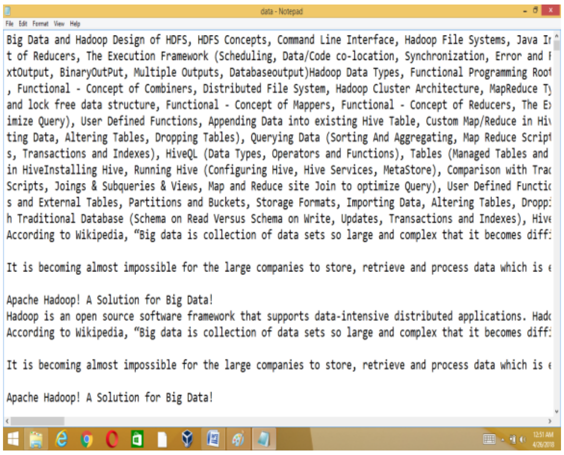
Fig 7 Semi Structured Dataset 2

C. Unstructured Dataset Description

1. Unstructured Dataset 1

The data which is considered as unstructured data is a plain text file. Analysis is performed over text file using Pig, Hive and Map Reduce.

Query: To count the occurrence of words



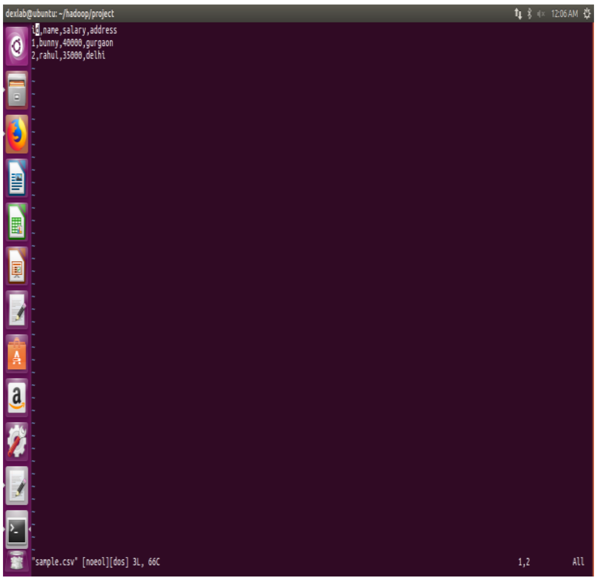
```
Big Data and Hadoop Design of HDFS, HDFS Concepts, Command Line Interface, Hadoop File Systems, Java Ir
t of Reducers, The Execution Framework (Scheduling, Data/Code co-location, Synchronization, Error and F
xtOutput, BinaryOutPut, Multiple Outputs, Databaseoutput)Hadoop Data Types, Functional Programming Root
, Functional - Concept of Combiners, Distributed File System, Hadoop Cluster Architecture, MapReduce Ty
and lock free data structure, Functional - Concept of Mappers, Functional - Concept of Reducers, The Ex
imize Query), User Defined Functions, Appending Data into existing Hive Table, Custom Map/Reduce in Hiv
ting Data, Altering Tables, Dropping Tables), Querying Data (Sorting And Aggregating, Map Reduce Script
s, Transactions and Indexes), HiveQL (Data Types, Operators and Functions), Tables (Managed Tables and
in HiveInstalling Hive, Running Hive (Configuring Hive, Hive Services, MetaStore), Comparison with Trac
Scripts, Joings & Subqueries & Views, Map and Reduce site Join to optimize Query), User Defined Functic
s and External Tables, Partitions and Buckets, Storage Formats, Importing Data, Altering Tables, Droppi
h Traditional Database (Schema on Read Versus Schema on Write, Updates, Transactions and Indexes), Hive
According to Wikipedia, "Big data is collection of data sets so large and complex that it becomes diff

It is becoming almost impossible for the large companies to store, retrieve and process data which is e

Apache Hadoop! A Solution for Big Data!
Hadoop is an open source software framework that supports data-intensive distributed applications. Hadc
According to Wikipedia, "Big data is collection of data sets so large and complex that it becomes diff

It is becoming almost impossible for the large companies to store, retrieve and process data which is e

Apache Hadoop! A Solution for Big Data!
```

Fig 8 Unstructured Dataset 1

2. Unstructured Dataset 2

The second sample of unstructured data is a Comma Separated File (CSV) containing details of an employee such as Name, Salary and address. A sample of this dataset is shown in Fig 9 below.

Query: To count the number of records



```
id,name,salary,address
1,bunny,40000,gurgaon
2,rahul,35000,delhi
"sample.csv" [noeol][dos] 3L, 66C                                   1,2           All
```

Fig 9 Unstructured Dataset 2

D. Implementation of Proposed System

The proposed system can be implemented by analyzing all the three formats of data i.e. Structured, Semi Structured and Unstructured on Pig, Hive and Map Reduce. Map Reduce is the programming paradigm of Hadoop which is written purely as Java Code. Pig is an abstraction layer over Map Reduce which is a scripting language. It reduces the programmer's effort. Hive also works over Map Reduce; it provides a SQL like Interface using which analytics can be done by querying the data. The three platforms are compared when data is analyzed to get same result.

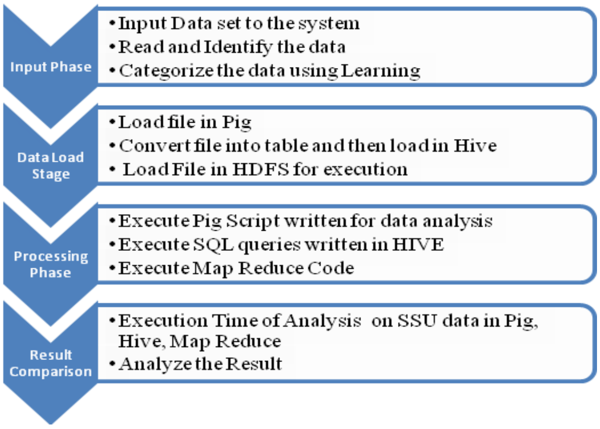

Input Phase
•Input Data set to the system
•Read and Identify the data
•Categorize the data using Learning
Data Load Stage
•Load file in Pig
•Convert file into table and then load in Hive
• Load File in HDFS for execution
Processing Phase
•Execute Pig Script written for data analysis
•Execute SQL queries written in HIVE
•Execute Map Reduce Code
Result Comparison
•Execution Time of Analysis on SSU data in Pig, Hive, Map Reduce
•Analyze the Result

Fig 10 Implementation Model for Hetergeneous Data Prei

III. Observation & Result

The execution time of each framework performing analysis on the three data formats are noted and maintained to infer the final results. The Table 1 & 2 below includes the time taken by each framework in analyzing all three data formats when data size is taken in MB's and GB's respectively.

Table 1. Comparison of Execution Time on different Big Data Platforms

Data Format	Execution Time of Hive (In Sec)	Execution Time of Pig (In Sec)	Execution Time of Map Reduce (In Sec)	Data size (In MB)
Structured Data Set 1	61.6	229.4	63.3	440.20
Structured Dataset 2	192.6	290.8	380.6	520.60
Semi Structured Dataset 1	193.3	180.3	208.4	374.94
Semi Structured Dataset 2	402.4	290.6	398.3	630.8
Unstructured dataset 1	300.8	180.6	232.3	493.45
Unstructured dataset	98.5	30.6	40.8	66.4

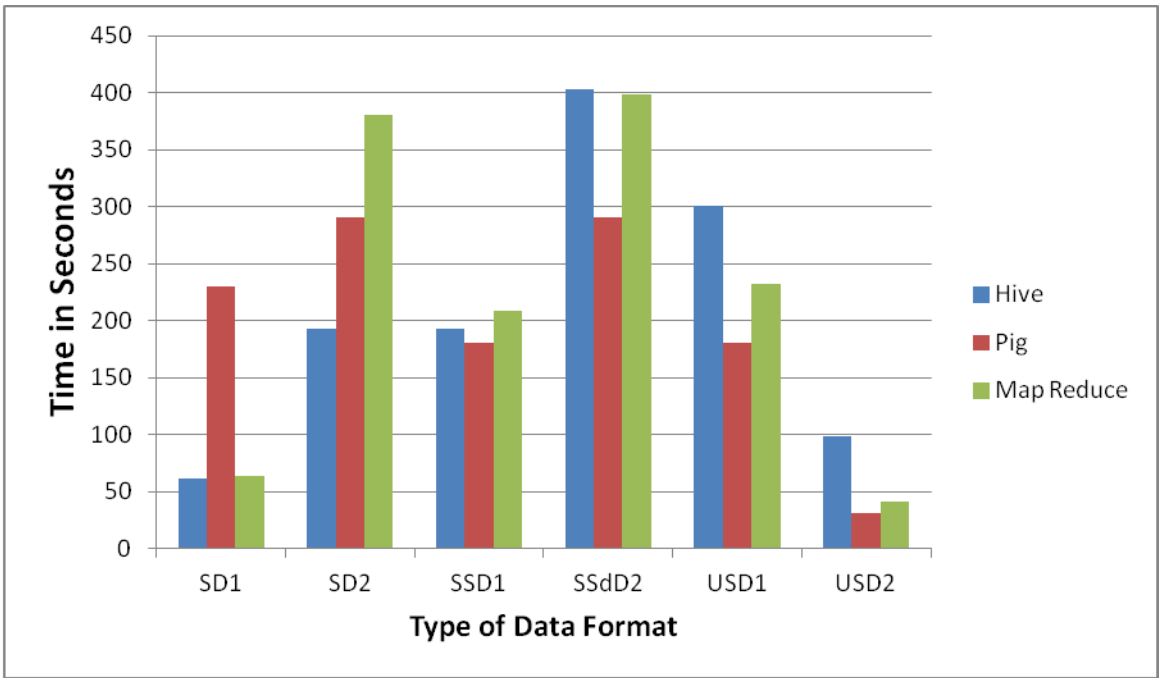


Fig 11 Comparison of Execution Time (Data Size in MB)

Table 2 Comparison of Execution Time on different Big Data Platforms

Data Format	Execution Time of Hive (In Sec)	Execution Time of Pig (In Sec)	Execution Time of Map Reduce (In Sec)	Data size (In GB)
Structured Data Set 1	138.6	516.2	158.25	1.32
Structured Dataset 2	288.9	436.2	608.9	1.1
Semi Structured Dataset 1	618.6	576.96	666.9	1.5
Semi Structured Dataset 2	603.75	435.9	637.28	1.26
Unstructured dataset 1	676.8	406.4	580.8	1.5
Unstructured dataset 2	1083.5	336.6	448.8	1.32

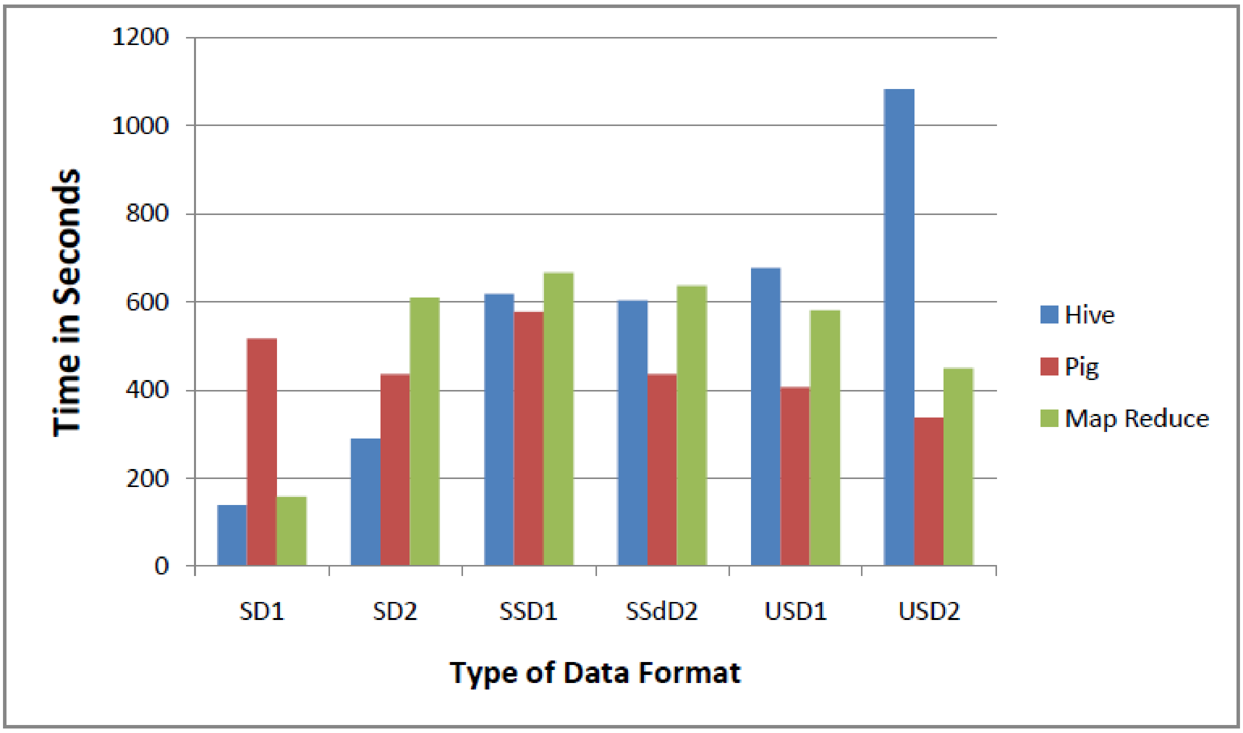


Fig 12 Comparison of Execution Time (Data Size in GB)

A. Analysis based on Variety of Data

1. Result Analysis of Structured Data

Data which is in its most organized form is termed as structured data. Data is maintained as rows and columns. In this study, two sets of structured data are taken into consideration. The size of Dataset 1 is 440.20 MB & 1.32 GB and that of dataset 2 is 520.60 MB & 1.1 GB. When analysis is performed over these two sets of structured datasets, it is found that Hive performs the fastest compared to other two. Since, the size of data is almost similar in both the cases. So it can be assumed that for structured data, Hive, Map Reduce can be taken as tool for analysis depending on the type of data.

2. Result Analysis of Semi Structured Data

Semi Structured data is the one which is neither organized nor unorganized. It contains a Meta data which can be treated as supporting data for the document. Some examples of Semi Structured data are Log files, JSON files, XML files etc. In this study two datasets of Semi Structured data are taken. Dataset 1 is an XML file and dataset 2 is a JSON file. The sizes of XML file are 374.94 MB & 1.5 GB. However, the sizes of JSON file 630.8 MB & 1.26 GB respectively. Analysis is performed over both the files and it is found that all the three platforms take similar time if the data size is small. However, as the size of data is increased, Pig performs considerably faster than other two.

3. Result Analysis of Unstructured Data

Lastly, analysis is performed over two sets of unstructured data. Data which is not organized at all such as text, images, video etc are the examples of unstructured data. A text file and a CSV file are taken as unstructured dataset. The result thus obtained after the analysis shows that Pig and Map Reduce give results almost at the same time. As the data size increases, again Pig shots a better performance than other two.

V. CONCLUSION

The paper explains Big Data Analytics performed by Hadoop. The performance of Pig, Hive and Map Reduce methodology is compared depending on time of execution for analyzing Big Data. Six data sets, two of each structured, semi structured and unstructured type are taken for analysis. Same queries are executed and the execution time is compared with data sets taken as MB's and GB's. The result observation infers that Hive shows best performance with Structured Data, Pig shows best with semi structured data. However, both Pig and Map Reduce shows good results with Unstructured data. These are few data sets considered for analysis, but big data analytics is purely dependent on type and size of data.

REFERENCES

1. Prof R A Fadnavis & Sannudhi Tabhane, "Big Data Processing using Hadoop", in IJCSIT, Vol I, 2015
2. Ashish Thusoo, Joydeep Sen Sarma, Namit Jain, Zheng Shao, Prasad Chakka, Ning Zhang, Suresh Antony, Hao Liu and Raghotham Murthy, "Hive – A Petabyte Scale Data warehouse using Hadoop", IEEE, 2010
3. Alan F. Gates, Olga Natkovich, Shubham Chopra, Pradeep Kamath, "Building a high level data flow system on top of Map Reduce : the Pig Experience", Proceedings of VLDB Endowment, Vol 2, Issue 2, August, 2009
4. Jeffery Dean & Sanjay Ghemawat, "Map Reduce : Simplified Data Processing on Large Clusters", 6th Symposium on Operating System Design and Implementation, Dec 2004
5. E. Laxmi Lydia & Dr. M. Ben Swarup, " Big Data analysis using Hadoop components like Flume, Map Reduce, Pig and Hive", IJCSET, Vol 5, Issue 11, Nov 2015
6. Bichitra Mandai, Ramesh Kumar Sahoo and Srinivas Sethi "Architecture of efficient word processing using Hadoop for Big Data Applications", in International Conference on Man and Machine Interfaccing",IEEE 2015
7. Poonam Vashisht and Vishal Gupta, "Big Data Analytics Techniques: A survey" , in IEEE 2015