ANALYZING VARIETIES OF STRUCTURED, SEMI-STRUCTUREDANDUNSTRUCTURED DATAUSING HIVEQL

[1]Parag C Shukla,[2]Dr. Kishor H. Atkotiya
[1]Assistant Professor& Head, [2]Professor
[1]Department of MCA, [2]Department of Statistics,
[1]Atmiya Institute of Technology & Science,Rajkot,India
[2]Saurashtra University, Rajkot, India

Abstract: Nowadays, many organizations is focusing on gathering and analyzing data. Analysis of data is necessary in today's world. Source of data is also not fixed; source of data can be in different form. It can be in structured manner like spread sheets, it can be in semi-structured manner like xml or it can be in unstructured manner like webpage, document or text. Analysis of this kind of diversified data is necessary. Hive Query Language can be used to analyze these varieties of data. This study will represent how hive can be used to analyze diversified data.

Index Terms **– Analyzing, Varieties, Structured, Unstructured, Semi-structured, Hive, HiveQL**

I. INTRODUCTION

Hive is a data warehousing tool which is used to process batch jobs on huge data that can be immutable. It is best suitable for data warehousing applications. It supports analysis of varieties of data. It is like SQL and supports rich data types like struct, array and Map. For faster access of data and for high performance, hive is supporting concept of partitioning. We can use static partitioning and dynamic partitioning. It has also functionality of bucketing instead of creation of thousands of partitions. Hive has a functionality to deal with semi-structured data, it supports SerDe which means serialization and deserialization. It has different kinds of file formats like Text File, Sequential File and RC File. Hive can be useful to analyze varieties of data.

II. ANALYSIS OF STRUCTURED DATA

For analysis of structured data, we should identify the source of structured data. It can be spreadsheets, OLTP systems or any RDBMS tool. Figure 1-1 shows the sources of structured data.

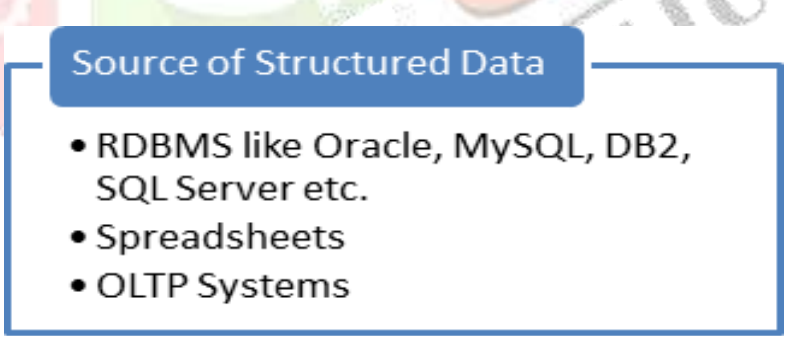

Figure 1-1 Source of Structured Data

Example:
Step-1 Create the data file which has structured data.

```
root@sandbox~
[root@sandbox ~]# cat data.csv
1001,RAM,DAA:AI:BDT:SEO:DTT,MARK1!60:MARK2!78:MARK3!55
1002,SITA,DAA:AI:BDT:JWT:DTT,MARK1!70:MARK2!79:MARK3!65
1003,LAXMAN,DAA:AI:BDT:JWT:DTT,MARK1!50:MARK2!72:MARK3!75
1004,URMILA,DAA:AI:BDT:SEO:DTT,MARK1!40:MARK2!73:MARK3!85
1005,BHARAT,DAA:AI:BDT:SEO:DTT,MARK1!80:MARK2!71:MARK3!95
[root@sandbox ~]#
```

Step-2 Create table in Hive

```
root@sandbox:~
hive> CREATE TABLE STUDENT_INFO (
    >                 RNO INT,
    >                 NAME STRING,
    >                 SUB ARRAY<STRING>,
    >                 MARKS MAP<STRING,INT>)
    > ROW FORMAT
    >     DELIMITED FIELDS TERMINATED BY ','
    >     COLLECTION ITEMS TERMINATED BY ':'
    >     MAP KEYS TERMINATED BY '!';
OK
Time taken: 0.7 seconds
hive>
```

Step-3 Load data in table from data.csv

```
hive> LOAD DATA LOCAL INPATH '/root/data.csv' OVERWRITE INTO TABLE STUDENT_INFO;
Copying data from file:/root/data.csv
Copying file: file:/root/data.csv
Loading data to table rnd.student_info
rmr: DEPRECATED: Please use 'rm -r' instead.
Moved: 'hdfs://sandbox.hortonworks.com:8020/apps/hive/warehouse/rnd.db/student_info' to trash
at: hdfs://sandbox.hortonworks.com:8020/user/root/.Trash/Current
Table rnd.student_info stats: [numFiles=1, numRows=0, totalSize=290, rawDataSize=0]
OK
Time taken: 1.781 seconds
hive>
```

Step-4 Verify data

```
root@sandbox~
hive> SELECT * FROM STUDENT_INFO;
OK
1001    RAM     ["DAA","AI","BDT","SEO","DTT"]  {"MARK1":60,"MARK2":78,"MARK3":55}
1002    SITA    ["DAA","AI","BDT","JWT","DTT"]  {"MARK1":70,"MARK2":79,"MARK3":65}
1003    LAXMAN  ["DAA","AI","BDT","JWT","DTT"]  {"MARK1":50,"MARK2":72,"MARK3":75}
1004    URMILA  ["DAA","AI","BDT","SEO","DTT"]  {"MARK1":40,"MARK2":73,"MARK3":85}
1005    BHARAT  ["DAA","AI","BDT","SEO","DTT"]  {"MARK1":80,"MARK2":71,"MARK3":95}
Time taken: 0.698 seconds, Fetched: 5 row(s)
hive>
```

Step-5 Analysis of Structured Data

```
root@sandbox~
hive> SELECT AVG(MARKS['MARK1']),SUM(MARKS['MARK1']),MAX(MARKS['MARK1'])
    > FROM STUDENT_INFO;
Total MapReduce CPU Time Spent: 1 seconds 830 msec
OK
60.0    300     80
Time taken: 20.103 seconds, Fetched: 1 row(s)
hive>
```

We can use different kinds of aggregate functions like avg, min, max, sum, count for analysis of data. In above data subject is stored as an array and marks are stored as a structure. We can retrieve individual records from table. In above retrieval you can check data of mark1, sum of mark1, maximum from mark1 and average of mark1 is analyzed.

III. ANALYSIS OF SEMI-STRUCTURED DATA

For analysis of semi-structured data, we should identify the source of semi-structured data. It can be XML, JSON or any other markup languages. Following figure shows the sources of semi-structured data.

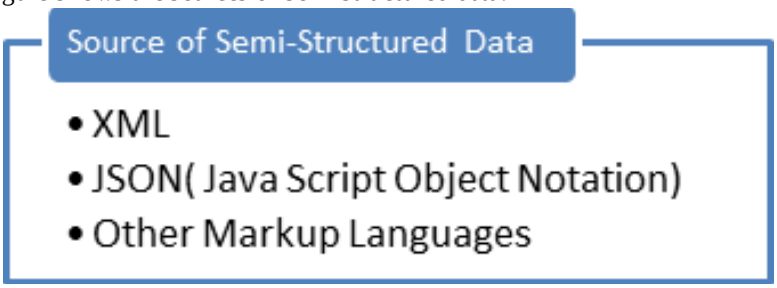

Figure 1-2 Source of Semi-Structured Data

Example:

Step-1 Create the data which has semi-structured data.

```
[root@sandbox ~]# cat input.xml
<student> <rollno> 1001 </rollno> <name> RAM </name> <city> Ayodhya </city> </student>
<student> <rollno> 1002 </rollno> <name> KRUSHNA </name> <city> Dwarika </city> </student>
<student> <rollno> 1003 </rollno> <name> NARENDRA </name> <city> Vadnagar </city> </student>
<student> <rollno> 1004 </rollno> <name> SACHIN </name> <city> Mumbai </city> </student>[root@sandbox ~]#
```

Step-2 Create table in hive.

```
root@sandbox:~
hive> CREATE TABLE XMLSAMPLE (xmldata STRING);
OK
Time taken: 1.011 seconds
hive>
```

Step-3 Load Data in table

```
hive> LOAD DATA LOCAL INPATH '/root/input.xml' OVERWRITE INTO TABLE XMLSAMPLE;
Copying data from file:/root/input.xml
Copying file: file:/root/input.xml
Loading data to table rnd.xmlsample
rmr: DEPRECATED: Please use 'rm -r' instead.
Moved: 'hdfs://sandbox.hortonworks.com:8020/apps/hive/warehouse/rnd.db/xmlsample' to trash at: hdfs://sandbo
x.hortonworks.com:8020/user/root/.Trash/Current
Table rnd.xmlsample stats: [numFiles=1, numRows=0, totalSize=359, rawDataSize=0]
OK
Time taken: 1.13 seconds
hive>
```

Step-4 Verify your data

```
root@sandbox:~
hive> SELECT * FROM XMLSAMPLE;
OK
<student> <rollno> 1001 </rollno> <name> RAM </name> <city> Ayodhya </city> </student>
<student> <rollno> 1002 </rollno> <name> KRUSHNA </name> <city> Dwarika </city> </student>
<student> <rollno> 1003 </rollno> <name> NARENDRA </name> <city> Vadnagar </city> </student>
<student> <rollno> 1004 </rollno> <name> SACHIN </name> <city> Mumbai </city> </student>
Time taken: 0.827 seconds, Fetched: 4 row(s)
hive>
```

Step-5 Analysis of Semi-Structured Data

```
hive> SELECT * FROM XMLSAMPLE
    > WHERE xpath_int(xmldata,'student/rollno')=1002;

Total MapReduce CPU Time Spent: 1 seconds 350 msec
OK
<student> <rollno> 1002 </rollno> <name> KRUSHNA </name> <city> Dwarika </city> </student>
Time taken: 12.198 seconds, Fetched: 1 row(s)
hive>
```

IV. ANALYSIS OF UNSTRUCTURED DATA

For analysis of unstructured data source can be web pages, images, free form text, audio, video, email, social media data, word documents etc. Following figure shows source of unstructured data.

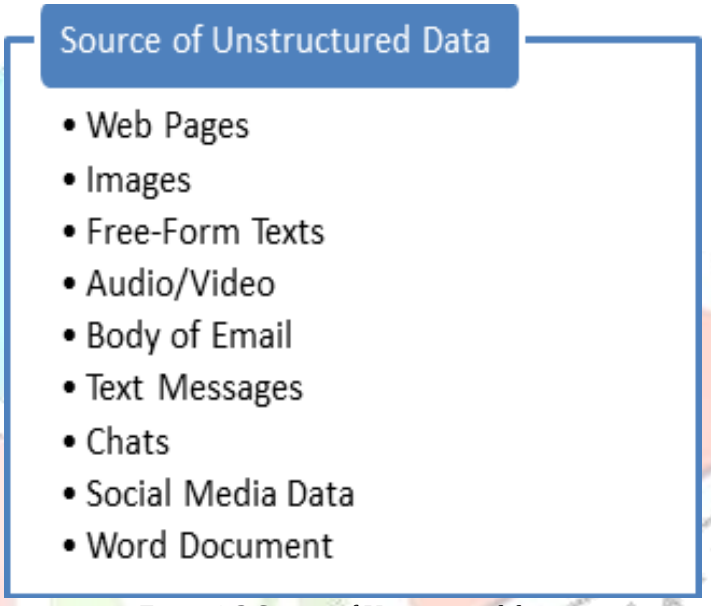

Figure 1-3 Source of Unstructured data.

Step-1 Create the unstructured data. Following file text file has unstructured data.

root@sandbox:~

```
[root@sandbox ~]# cat sample.txt
Hadoop is easy
Hadoop is cool
Hadoop is simple robust scalable and mostly used.
Hadoop uses Java
NoSQL and Hadoop both are widely used.[root@sandbox ~]#
```

Step-2 Create the table and load data in hive

root@sandbox:~

```
hive> CREATE TABLE docs(line STRING);
OK
Time taken: 0.979 seconds
hive> LOAD DATA LOCAL INPATH '/root/sample.txt' OVERWRITE INTO TABLE docs;
OK
Time taken: 1.145 seconds
hive>
```

Step-3 Verify your data

```
root@sandbox:~
hive> SELECT * FROM DOCS;
OK
Hadoop is easy
Hadoop is cool
Hadoop is simple robust scalable and mostly used.
Hadoop uses Java
NoSQL and Hadoop both are widely used.
Time taken: 0.793 seconds, Fetched: 5 row(s)
hive>
```

Step-4 Count the occurrences of similar words

```
root@sandbox:~
hive> SELECT WORD, COUNT(1) AS LEN
    > FROM
    >         (SELECT explode (split (line,' ')) AS WORD
    >          FROM docs) tmp
    > GROUP BY WORD
    > ORDER BY LEN DESC;
OK
Hadoop  5
        3
is      3
and     2
used.   2
widely  1
uses    1
simple  1
scalable        1
robust  1
mostly  1
cool    1
both    1
are     1
NoSQL   1
Java    1
easy    1
Time taken: 42.63 seconds, Fetched: 17 row(s)
hive>
```

Above is example to count similar kinds of word from unstructured data file.

V. CONCLUSION

Hive Query Language is very useful tool to analyze varieties of data. We can analyze structured data that can be in OLTP system, spreadsheets or in any RDBMS tables. We can analyze semi-structured data like XML, JSON or any markup language. We can also analyze unstructured data like document, weblogs, email or social media data using hive. As we know analysis of data is very important nowadays, it is prime requirement of many organizations to analyze the data to take better and efficient decision.

REFERENCES

[1] From the Gartner IT Glossary: What is Big Data? [Online Resource] https://research.gartner.com/definition-whatis-big-data?resId=3002918&srcId=18163325102
[2] Apache Hive, https://hive.apache.org/
[3] Tutorial Hive, https://cwiki.apache.org/confluence/display/Hive/Tutorial
[4] Accessing Hadoop data using Hive https://cognitiveclass.ai/courses/hadoophive/
[5] Big Data and Analytics – Wiley Publication, Seema Acharya, Subhashini Chellapan