**Analysis on H1-B Visa Petitions using Big Data Technologies**

|  |  |
| --- | --- |
| **CWID** | **Group Member Name** |
| A20430008 | SIDDHI SANJAY KULKARNI |
| A20429199 | SHIVANGEE DURGADAS KULKARNI |

CSP 554 Big Data Technologies

**Professor:** Adam McElhinney

November 25th, 2019

**Introduction:**

Now-a-days there is a variety of the information available from different websites. It is becoming very difficult to handle this data and so we need to process this data. The traditional data processing techniques are not capable of handling this huge amount of data and so there is a need for the Big Data technologies to handle and process this data. The tools like Hadoop, Map Reduce, etc. and other tools are capable of handling this data. [1] In this project we have selected the H1-B data of the employees and are going to analyze this data. We have used some of the test cases for analyzing the data in Hive, Pig, Spark, and Map Reduce. Also, we will compare the query execution time for the queries executed in Hive, Pig and Spark SQL.

**Map Reduce:**

**Map Reduce** – Map reduce is a software framework for processing of data in a distributed fashion over different machine. The core idea behind map reduce is mapping your data set into a collection of <key, value> pairs and then reducing overall pairs with the same key.[2]

Map Reduce program works in two phases:

**1.** Map phase (Splits and Mapping)

**2.** Reduce phase (Shuffling and Reducing)

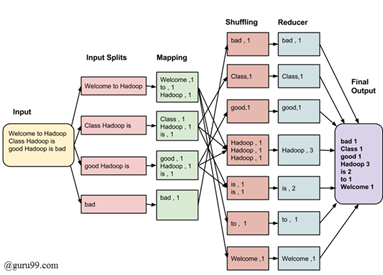


Fig.1: Architecture Of Map Reduce [2]

The data goes through the following phases:

Input Split: An input is split into fixed sized pieces called input split. It is the chunk of the input.

Mapping: This is the first step of the execution of a map-reduce program. The data is passed to the mapper function to generate the key,value pairs.

Shuffling: This phase takes the input of the mapping phase and consolidates the relevant records from the map phase.

Reducing: The output from the shuffling phase are aggregated.

The complete Map Reduce architecture is controlled by two entites:

1. Job Tracker- Acts like a master.
2. Multiple Task Tracker- Acts like a slave.

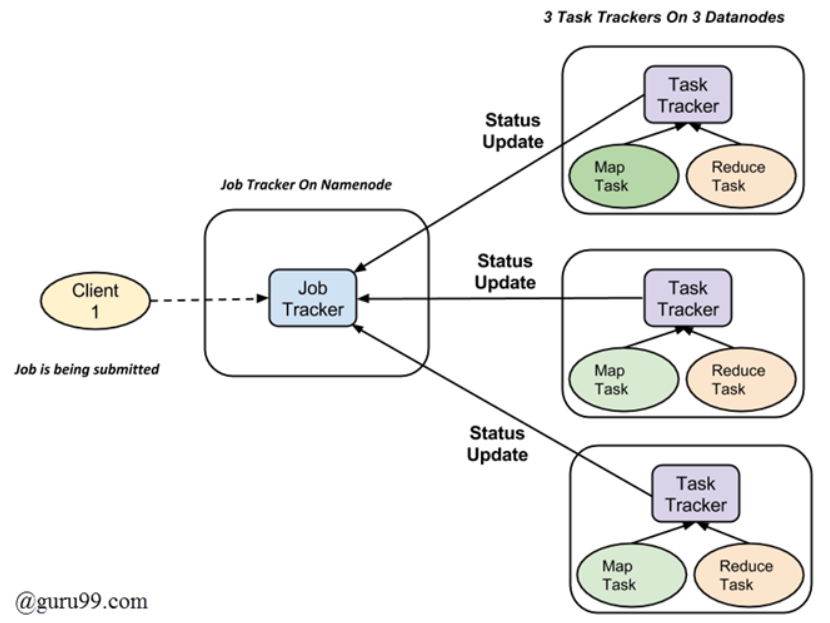


Fig.2: Job Execution in Map Reduce

The Job tracker resides on the Name node and the Task Tracker resides on the Data Nodes.

1. A job is divided into multiple tasks which are then run onto multiple data nodes in a cluster.
2. The job tracker is responsible to coordinate the activity by scheduling the tasks to run on different data nodes.
3. Execution of individual task is then looked by the task tracker.
4. Task tracker sends the heartbeat signal to the job tracker and the task completion report to the Job Tracker to notify him the current state of the system.[2]

Advantages:

1. Hadoop is a platform that is highly scalable. It is highly scalable.
2. MapReduce supports parallel programming. It divides tasks and executes parallely.

Disadvantages:

1. Mapreduce cannot be used in Real-time processing.
2. Mapreduce cannot be used in OLTP.

**Apache Hive:**

Apache Hive is an open source data warehouse system built on top of Hadoop Housed for querying and analyzing large datasets stored in Hadoop files. It processes structured and semi-structured data in Hadoop. Hive requires one additional component known as Metastore which is not present in Hadoop. It stores metadata for each of the tables like their schema and location. Hive also includes the partition metadata. [3]

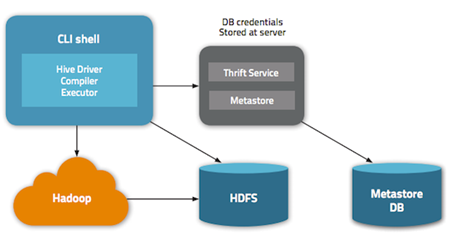


Fig.3: Architecture Of Hive

The architecture of Hive includes the following components:

1. Metastore- It stores the metadata for each of the tables like schema and location. It also includes partition metadata.
2. Driver- It acts like a controller for the execution of the HiveQL statements by creating sessions. It monitors the lifecycle and the progress of the query execution.
3. Compiler- It compiles the Hive query. It converts the query to an execution plan. The compiler converts the query to an Abstract Syntax Tree (AST). Then the AST is converted to a Directed Acyclic Graph (DAG).
4. Optimizer- It performs the various transformations on the execution plan to provide optimized DAG. The optimizer can perform the operations like splitting the tasks and then applying the transformation on data before performing the reduce operation.
5. Executor- It is responsible for the execution of tasks.[3]

Advantages:

1. Hive helps querying larger datasets residing in distributed storage.
2. Multiple users can simultaneously query the data using Hive-QL.
3. Data extract/transform/load can be done easily.

Disadvantages:

1. It is not designed for Online Transaction Processing (OLTP), it is only used for Online Analytical Processing (OLAP).
2. Hive does not support Sub-queries.

**Apache Pig:**

Pig is a high-level programming language useful for analyzing large data sets. A pig was a result of development effort at Yahoo! It enables focus on analyzing the bulk data sets and reduce the time for the execution of the Map Reduce programs.

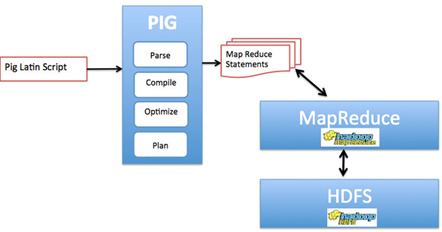


Fig.4: Architecture Of Pig

The components of the Pig architecture are as follows:[5]

Parser- It performs the syntax check of the script and the other miscellaneous checks. The output of the parser will be a DAG.

Optimizer- The DAG is passed on to the logical optimizer for carrying out the logical operations like projection and pushdown.

Compiler- It compiles the logical plan into a series of MapReduce jobs.

Execution Engine- The map reduce jobs are submitted to Hadoop in a sorted order. The jobs are executed on Hadoop producing the desired results.[4]

Advantages:

1. It is a dataflow structure.
2. Pig is effective for unstructured.
3. Pig uses lazy evaluation.

Disadvantages:

1. Pig delays execution. The final result is displayed only after final command is executed.
2. In Apache pig, data schema is enforced implicitly. This is the disadvantage of pig.

**Apache Spark:**

Apache Spark is an open source, Hadoop-compatible, fast and general-purpose cluster computing platform. It is a powerful open-source engine that provides real-time stream

processing, interactive processing, graph processing, in-memory processing as well as batch processing with very fast speed, ease of use and standard interface.[5]

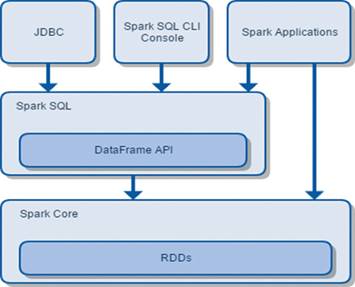


Fig.5: Architecture Of Apache Spark

The architecture of Spark includes the following components:

1. Spark Core- It is a kernel of the Spark which provides an execution platform for all the Spark applications. Resilient Distributed Dataset (RDD) is a fundamental unit of Apache Spark which is a collection of elements across cluster nodes that can perform parallel operations. The RDDs are immutable but can generate a new RDD by transforming existing RDD. There are three ways to create RDD in Spark:

1. Parallel Collections- We can create this RDD by invoking the parallelize method in driver program.

2. External datasets- One can create RDD by calling the textFile method. This method takes the URL of a file and reads it as a collection of lines.

3. Existing RDDs- By applying transformation on the existing RDD.

1. Spark SQL- It enables the users to run the SQL/HQL queries on the Spark. It allows us to process the structured as well as unstructured data.

Advantages:

1. It is easy to use also speed of SPARK is better than other tools like Apache hive and pig.
2. Advanced analytics is used in spark.
3. Apache spark is dynamic in nature.

Disadvantages:

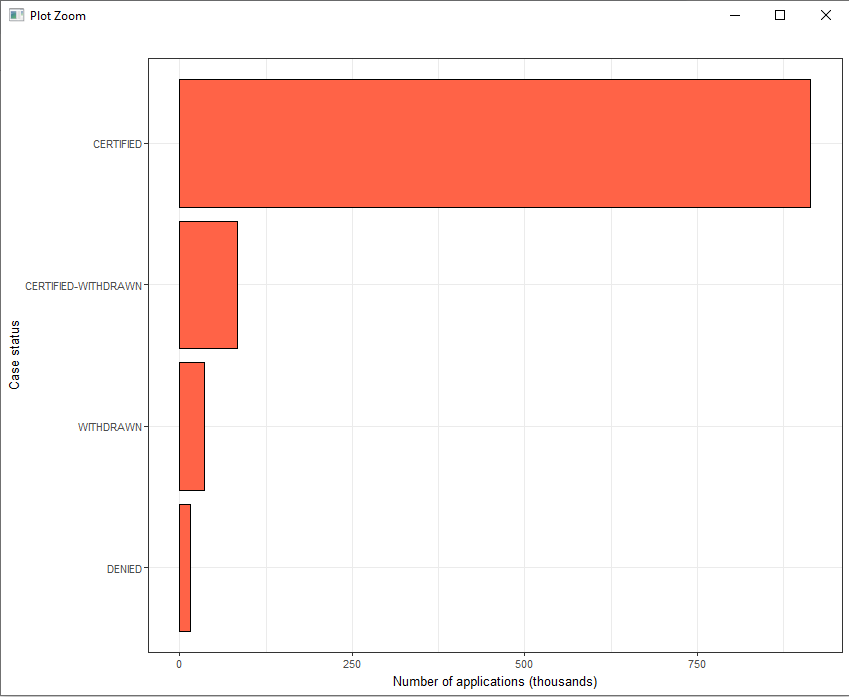
1. There is no automatic optimization process in Apache Spark.
2. Apache spark is not a suit for a multiuser environment.

**Implementation:**

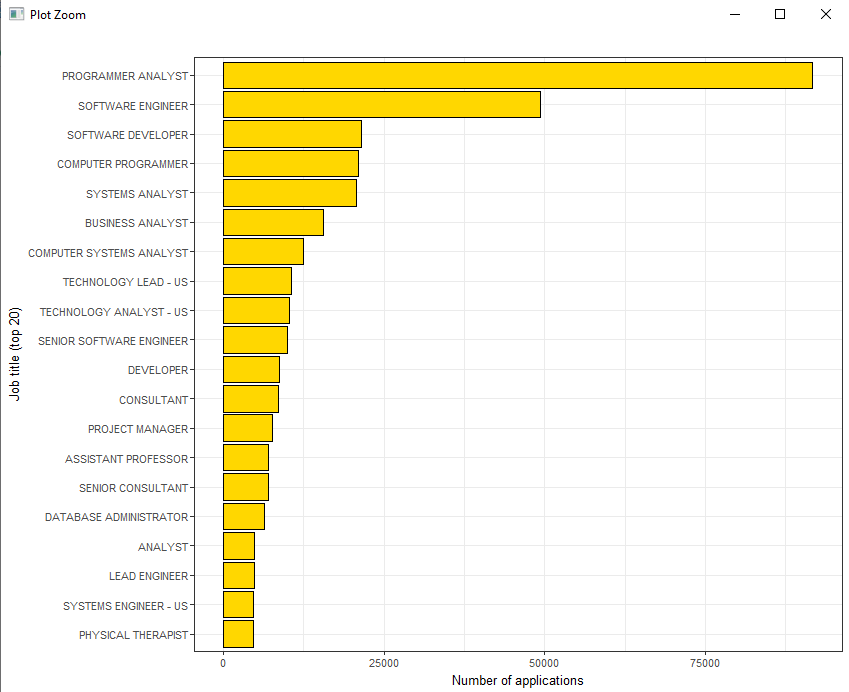
**Step1: Understanding dataset [6]**:

* **The meaning of the columns are as follows**:
* Case\_status – Status of the application
* Employer\_name – Name of employer registered in H1B visa application
* Soc\_name – Occupation code for employment
* Job\_title – Job title for employment
* Full\_time\_position – Whether application is for full-time position or part-time position
* Prevailing\_wage – The most frequent wage
* Year – The application year
* Worksite – The address of the employer worksite
* **Data Visualization**:

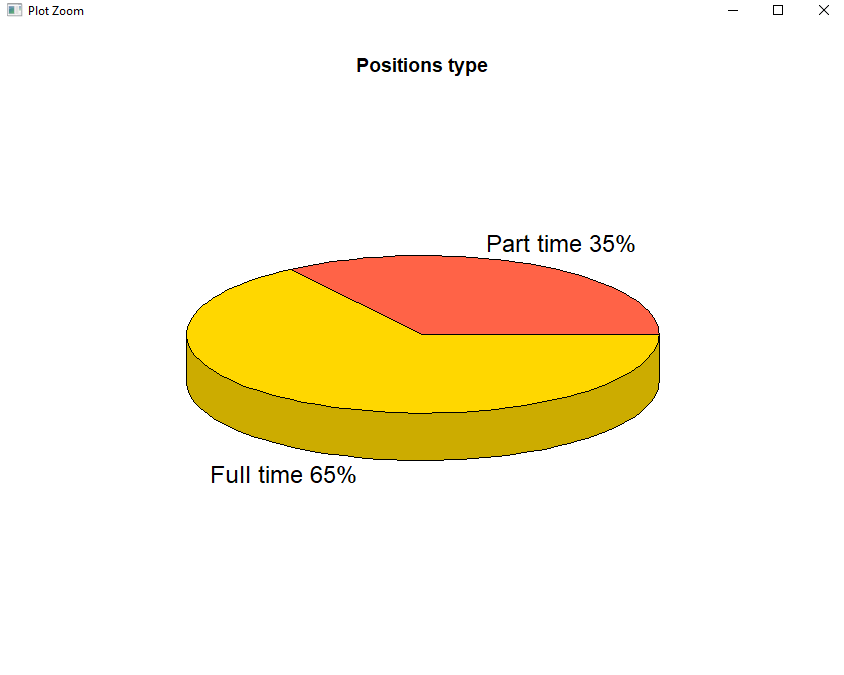
1. Distribution of CASE\_STATUS:



1. Top 20 of most frequent job titles in visa applications:

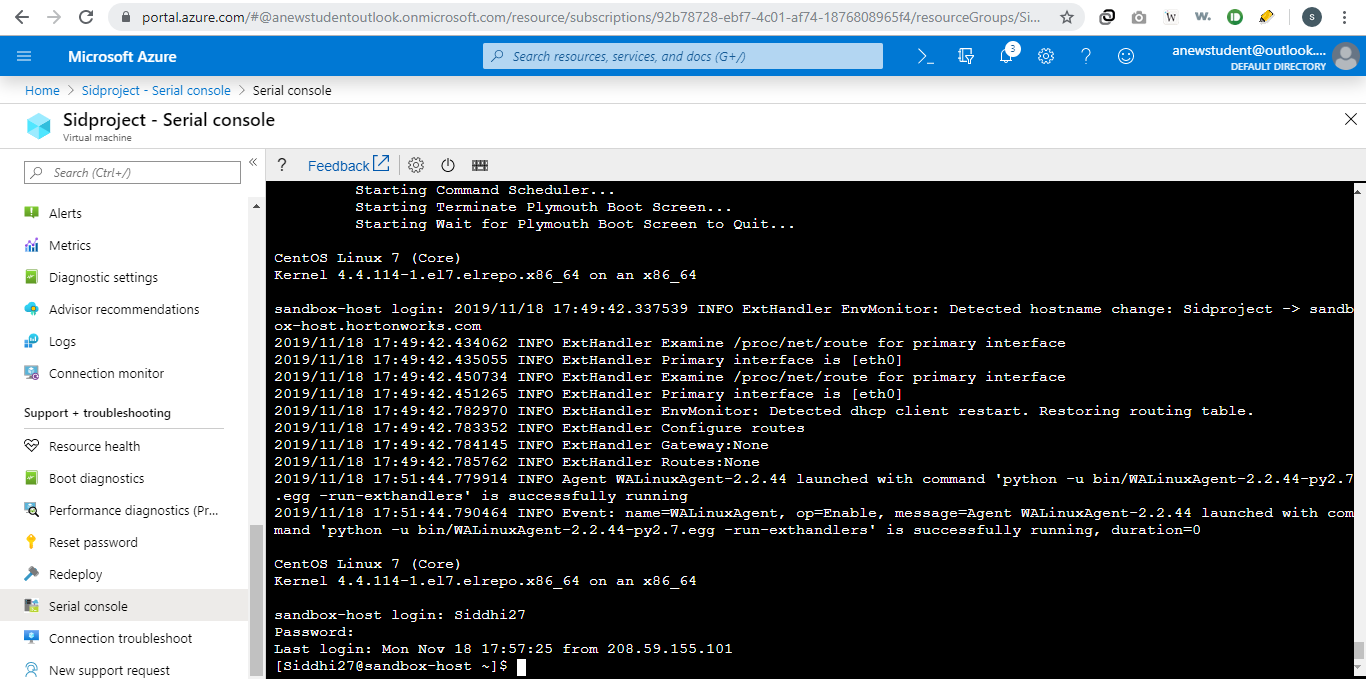


1. Percent of full-time positions from the total number of applications:



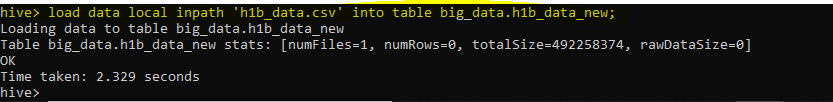
**Step 2:** Step by step execution

1. **Starting the virtual machine:**

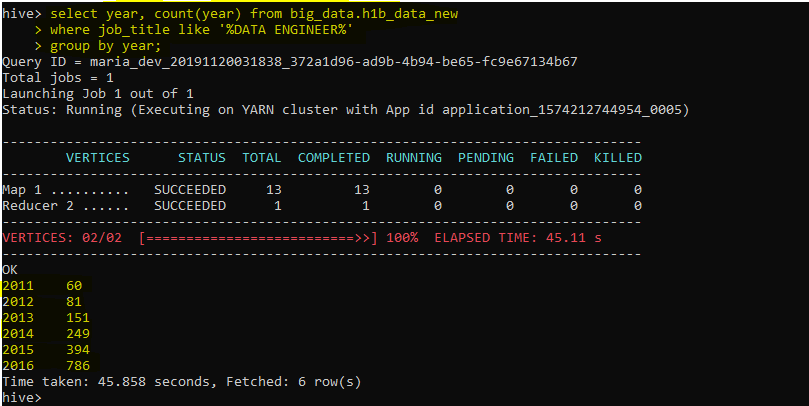


1. **Starting with Hive:**

Load the data in Hive

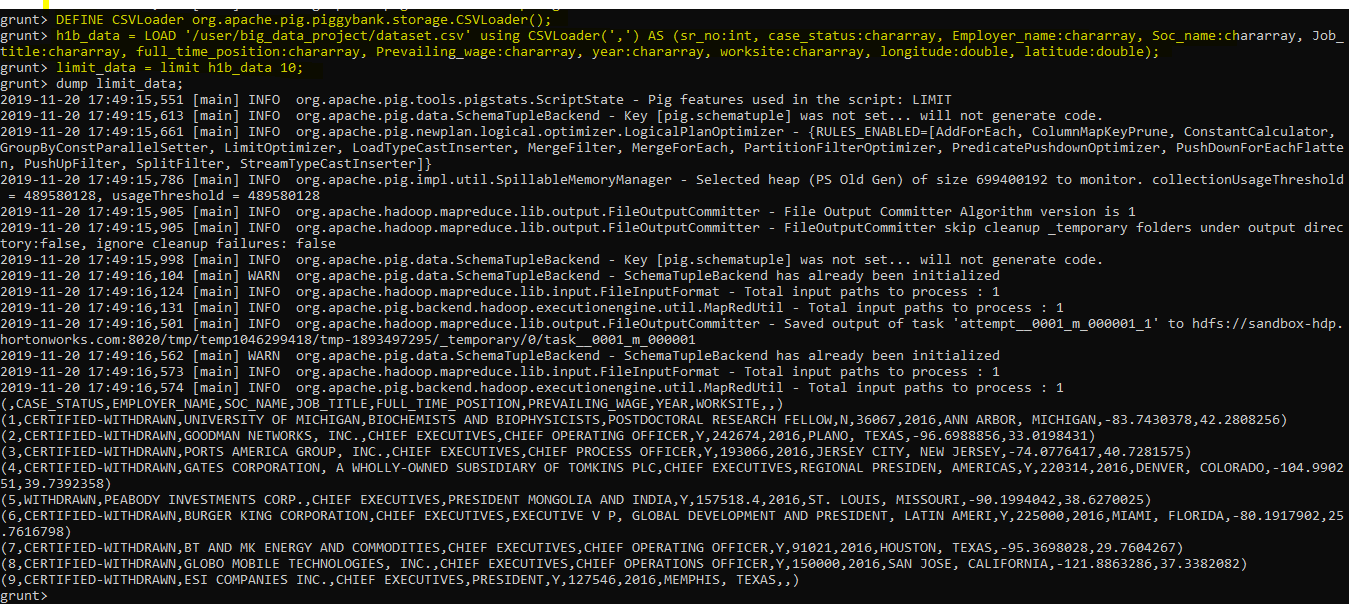


Is number of petitions with Data Engineer job title increasing over time?

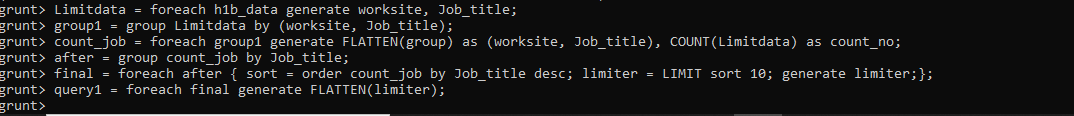


1. **Starting with Pig:**

Load data in Pig



Which worksite has popular job title?

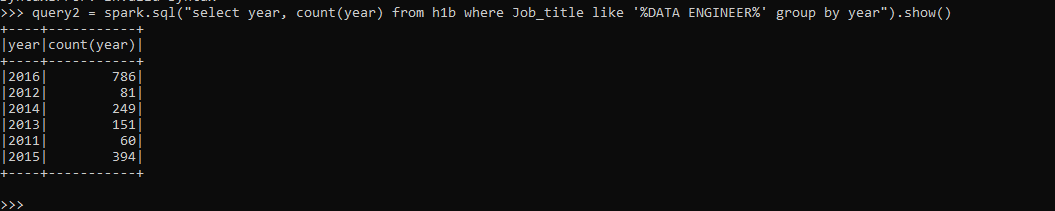


1. **Starting with Spark:**

Load data in Spark SQL



Is number of petitions with Data Engineer job title increasing over time?



**Experimentation with Hive, Pig and Spark processing environment:**

The amount of time consumed during input a user query for finding records from the Hive technique is termed here as the query execution time. In order to measure the query execution time, below listed queries are fired on the Hive interphase and their performance is observed. After completing the observations first time for all the queries.

Queries:

1)    Which worksite has the popular job title?

2)    Is the number of petitions with Data Engineer job title increasing over time?

3)    Which part of US has most Data Engineer jobs?

4) Which industry has the greatest number of Data Scientist positions?

5)    Which employers file the most petitions each year?

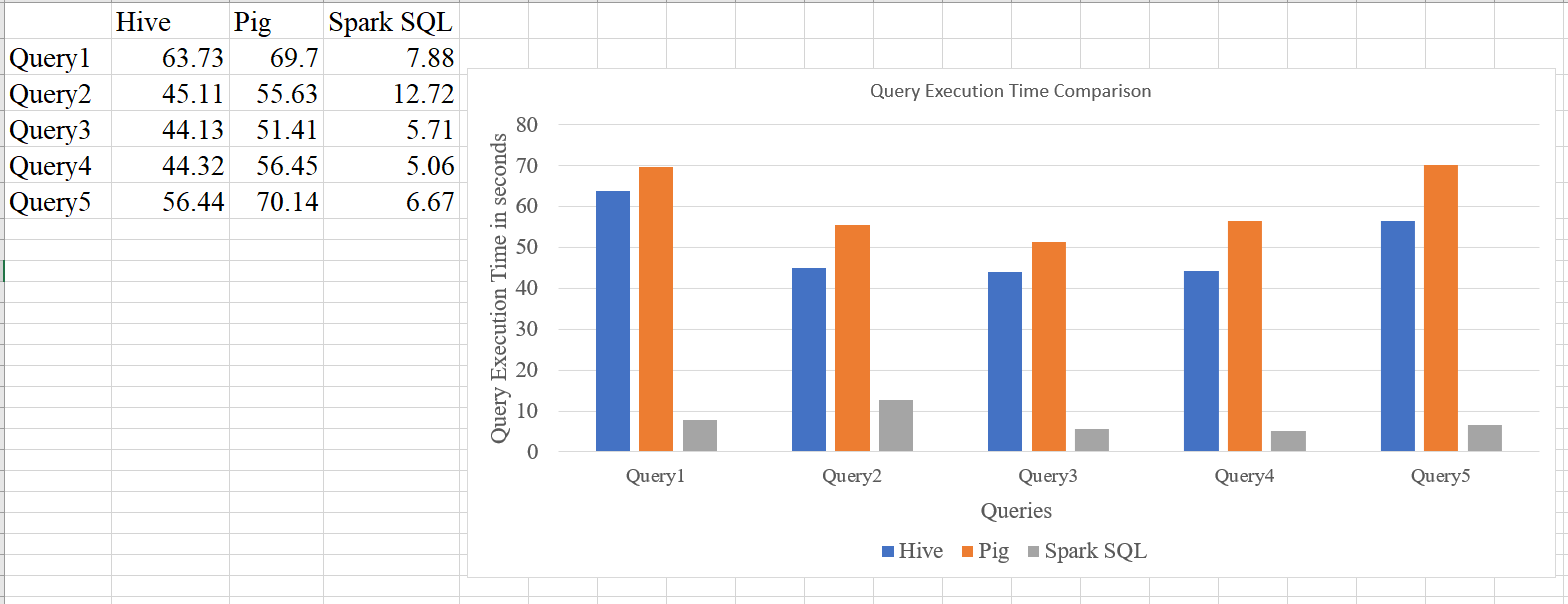
6)   What is the general trend of the employer’s H1-B status in the period 2011-16?

7)    What is the general trend of the employer name with respect to the year?

**Results and Discussion:**

**Compare performance of Hive, Pig and Spark**

From the execution of the queries mentioned above we conclude that Spark is more efficient than Hive and Pig. This is so because Spark uses less time for execution of queries, less lines of code and the structure of the output is easy to understand than Hive and Pig. The screenshot below displays the query execution time of each tool and its pictorial view of the comparison.



**Conclusion:**

Thus, the results represent that the performance of the Spark SQL is better than Hive and Pig. Spark is promising of enhanced performance which is 100 times faster than Hadoop MapReduce for some applications. Spark also supports the low-latency in-memory data processing capability which is missing in the Map Reduce.

**References:**

[1].   Monu, Monika and Pal, Sat, Simulation of Performance Analysis of MongoDB, PIG, HIVE Storage, Map Reduce, Spark and Yarn (April 4, 2019). Proceedings of International Conference on Sustainable Computing in Science, Technology and Management (SUSCOM), Amity University Rajasthan, Jaipur - India, February 26-28, 2019. Available at SSRN: https://ssrn.com/abstract=3365403 or <http://dx.doi.org/10.2139/ssrn.3365403>

[2] <https://www.guru99.com/introduction-to-mapreduce.html>

[3] <https://data-flair.training/blogs/apache-hive-tutorial/>

[4] <https://www.tutorialspoint.com/apache_pig/apache_pig_architecture.htm>

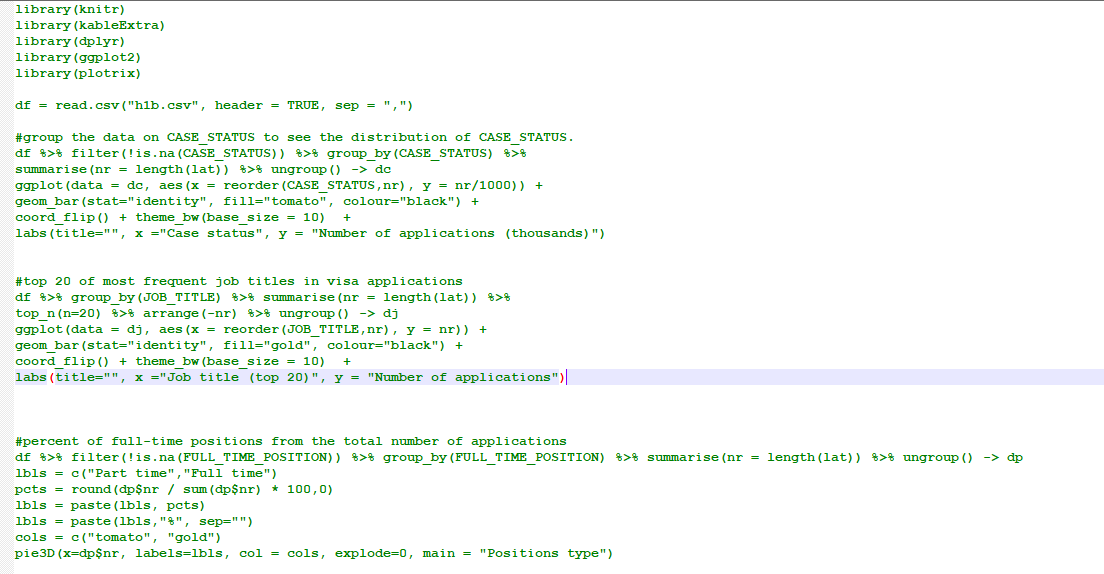
[5] <https://data-flair.training/blogs/what-is-spark/>

Dataset:

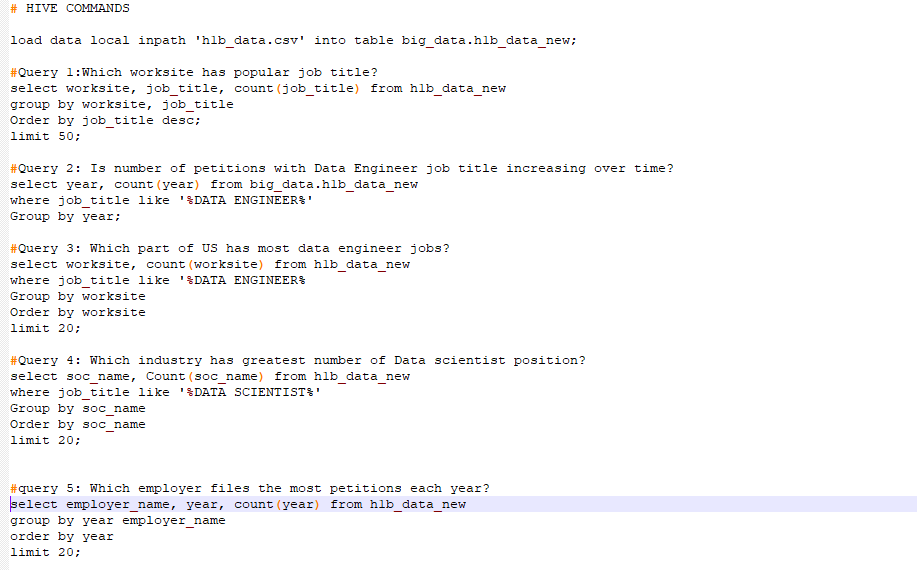
[6] H1B dataset (2011-16). Retrieved from <https://www.kaggle.com/nsharan/h-1b-visa>

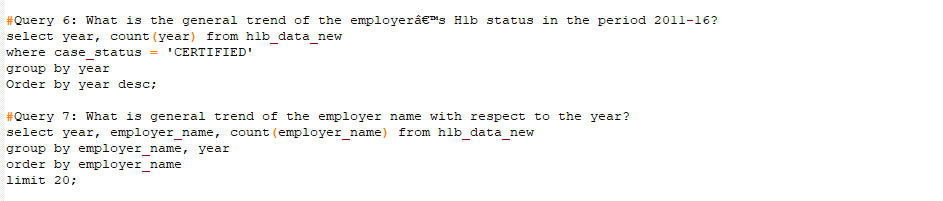
**Appendix:**

1. Data understanding and data visualization Code in R:

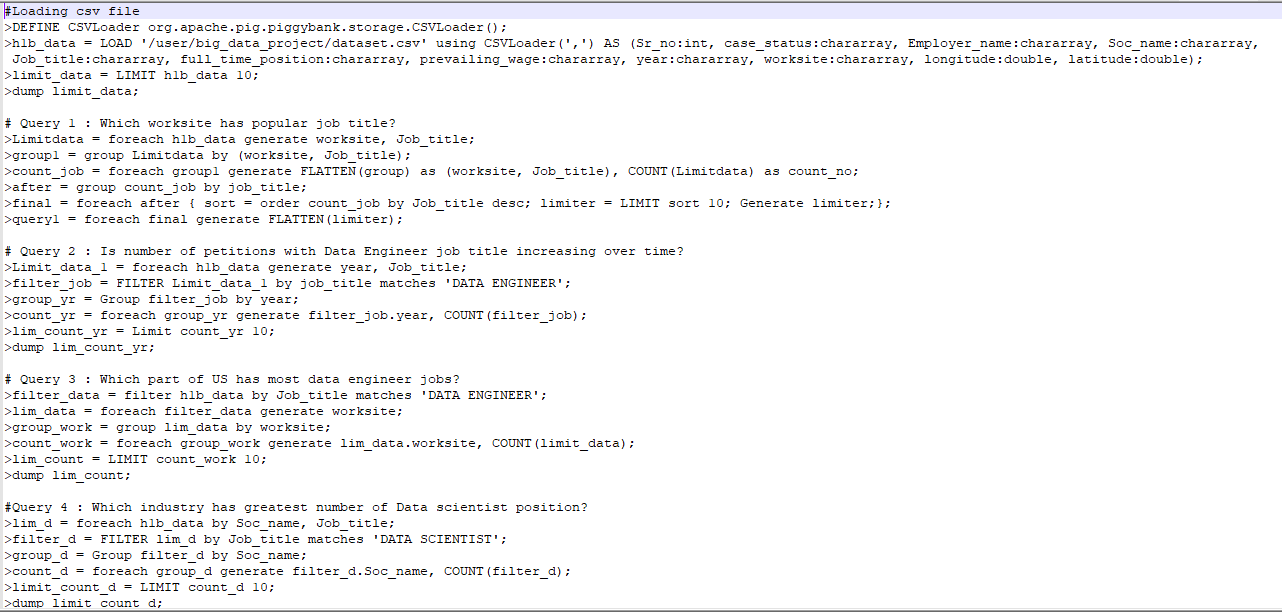


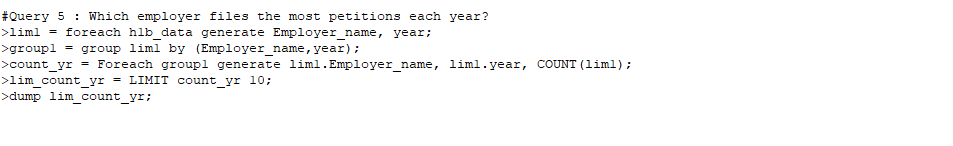
1. Hive commands:



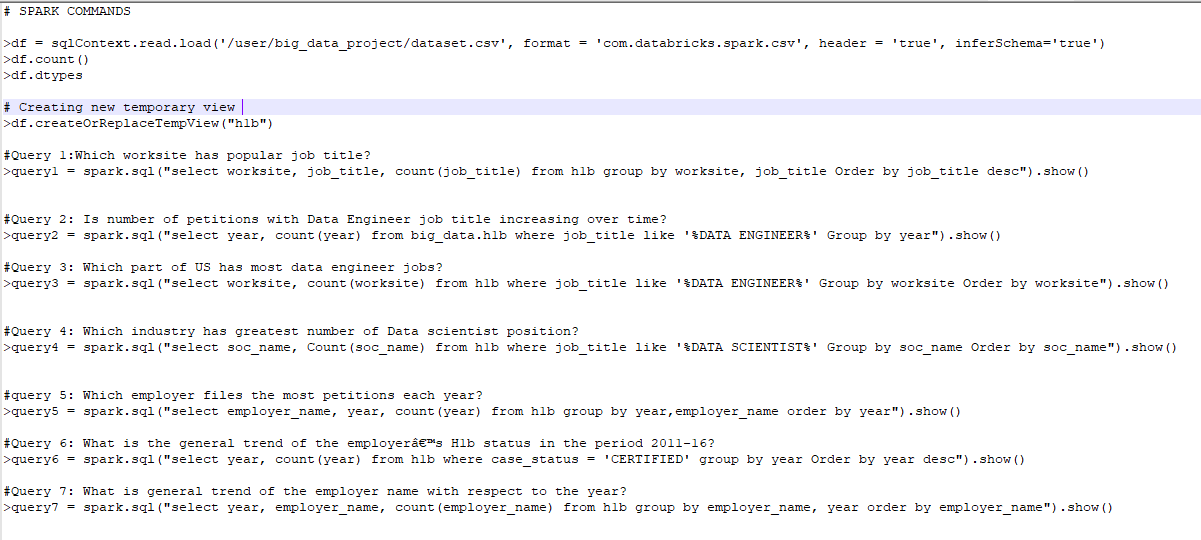


1. Pig commands:

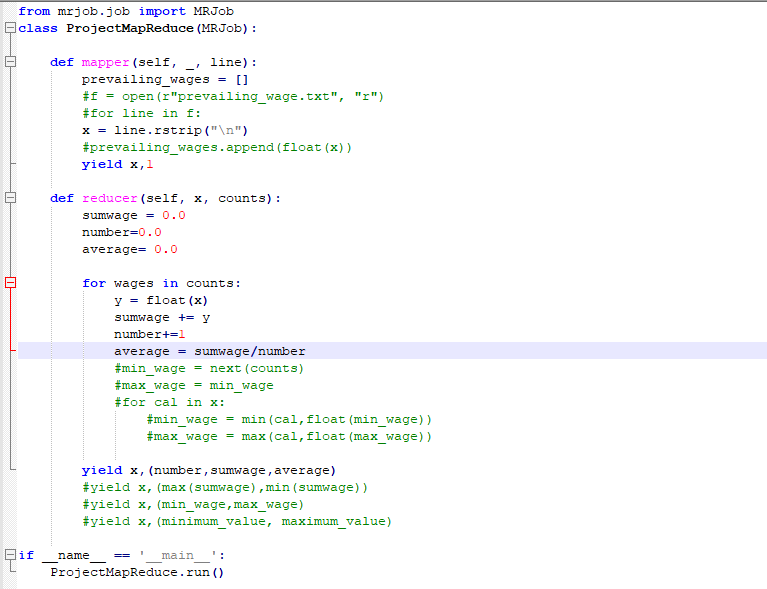




1. Spark SQL commands:



1. Map Reduce:



Git hub link: <https://github.com/Siddhikul27/Big-Data-Project>

Video Demo link: <https://drive.google.com/drive/u/1/folders/1o3JzB2tuKN_avIhvQSN92znubzNnYwdE>