**Analyzing H1B Applications**

**Using Hadoop**

Presented By:

Rahul Sihag

S.ID: S181107200294

NIIT, Gurugram.

**ABSTRACT**

**BIG DATA**

Big data is a term that describes the large volume of data – structured (Relational data), semi-structured (XML data) and unstructured (Word, PDF, Text, Media Logs.) data, that overrun a business on a day-to-day basis. But it’s not the amount of data that’s important. It’s what organizations do with the data that matters.

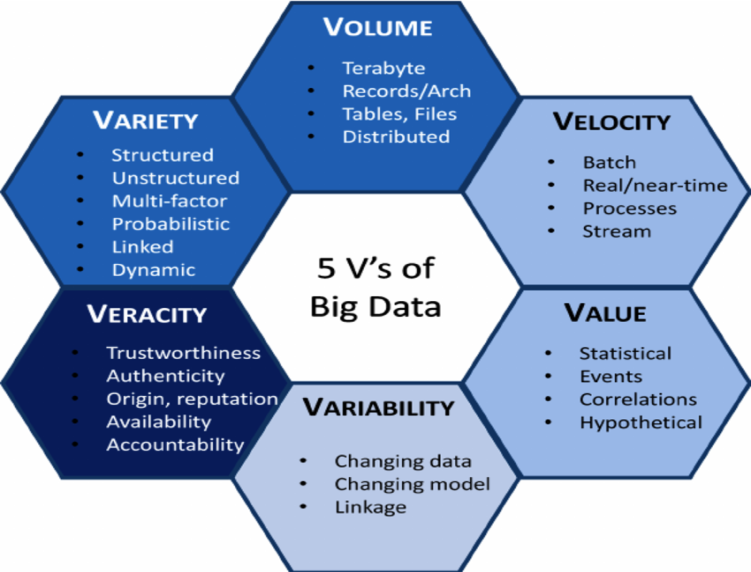


Figure 1: 5 V’s of Big data.

**Hadoop**

Hadoop is an open-source software framework for storing data and running applications on clusters of commodity hardware. It provides massive storage for any kind of data, enormous processing power and the ability to handle virtually limitless concurrent tasks or jobs.

**Hadoop History**

As the World Wide Web grew in the late 1900s and early 2000s, search engines and indexes were created to help locate relevant information amid the text-based content. In the early years, search results were returned by humans. But as the web grew from dozens to millions of pages, automation was needed. Web crawlers were created, many as university-led research projects, and search engine start-ups took off (Yahoo, AltaVista, etc.).

One such project was an open-source web search engine called Nutch – the brainchild of Doug Cutting and Mike Cafarella. They wanted to return web search results faster by distributing data and calculations across different computers so multiple tasks could be accomplished simultaneously. During this time, another search engine project called Google was in progress. It was based on the same concept – storing and processing data in a distributed, automated way so that relevant web search results could be returned faster.

In 2006, Cutting joined Yahoo and took with him the Nutch project as well as ideas based on Google’s early work with automating distributed data storage and processing. The Nutch project was divided – the web crawler portion remained as Nutch and the distributed computing and processing portion became Hadoop (named after Cutting’s son’s toy elephant). In 2008, Yahoo released Hadoop as an open-source project. Today, Hadoop’s framework and ecosystem of technologies are managed and maintained by the non-profit Apache Software Foundation (ASF), a global community of software developers and contributors.

**Why hadoop is important?**

* **Ability to store and process huge amounts of any kind of data, quickly.** With data volumes and varieties constantly increasing, especially from social media and the Internet of Things (IoT), that's a key consideration.
* **Computing power.** Hadoop's distributed computing model processes big data fast. The more computing nodes you use the more processing power you have.
* **Fault tolerance.** Data and application processing are protected against hardware failure. If a node goes down, jobs are automatically redirected to other nodes to make sure the distributed computing does not fail. Multiple copies of all data are stored automatically.
* **Flexibility.** Unlike traditional relational databases, you don’t have to pre-process data before storing it. You can store as much data as you want and decide how to use it later. That includes unstructured data like text, images and videos.
* **Low cost.** The open-source framework is free and uses commodity hardware to store large quantities of data.
* **Scalability.** You can easily grow your system to handle more data simply by adding nodes. Little administration is required.

**Hadoop Glossary**

**Hadoop common -** The libraries and utilities used by other hadoop modules.

**Hadoop Distributed File System (HDFS) –** Java based scalable system that stores data across multiple machines without prior organization.

**Yarn**-(yet another resource negotiator) provides resource managements for the process running on hadoop.

**Map Reducer**- A parallel processing software framework. It is comprised of two steps. Map step is a master node that takes input and partitions them into smaller sub-problems and then distributes them to worker nodes. After the map step has taken place, the master node takes the answer to all of the sub-problems and combines them to produce output.

**Hadoop components used in this project**

**Hive**: a data warehousing and SQL like query language that presents the data in the form of tables. Hive programming is similar to data warehousing.

**Pig**: a platform for manipulating data stored in HDFS and that includes a compiler for map reduce programs and high level language called Pig Latin.it provides a way to perform data extractions, transformation and loading and basic analysis without having to write MapReduce programs.

**Sqoop**: a connection and transfer mechanism that moves the data between hadoop and relational databases.

**ZooKeeper:** an application that coordinates distributed processing.

**Acknowledgement**

I have relied on many people, both directly and indirectly, for this project.

I would like to thank Mr. Sandeep Aggarwal, from whom I have learned, and continue to learn Hadoop.

I am particularly grateful to Mr. Naveen Goyal (Tech. Mentor) for his encouragement, support, and friendship.

Thanks also go to the many others with whom I have had conversations or email discussions over the project.

I want to thank my colleagues for being incredibly supportive in allowing me the time to work and to get it finished promptly.

**Project Outline**

|  |  |
| --- | --- |
| Title | Analysing H1B Applications Using Hadoop |
| Input | h1b.csv (h1b visa data) |
| Data Elements | S\_no, case\_status, employer\_name, soc\_name, job\_title, full\_time\_position, prevailing\_wage, year, worksite, longitute, latitute. |
| Purpose | Performing analysis on the H1B visa applicants between the years 2011-2015, to identify the growth cycle is increasing or not, which industry has granted more visa’s etc. |

**Project Implementation**

**Assumptions:**

1. Hadoop is running.

2. Ecosystem products (hive, pig, mysql) are installed

3. H1b visa data is available on HDFS in the text format.

**Prerequisites:**

The h1b visa data in csv format hence then needed to be converted in text file format in Hadoop file system.

**Steps for conversion:**

Step 1: create a table in hive to read entire record as one text string.

Step 2: create a table to convert and store csv string into different fields which will be create a file on hadoop file system in text format.

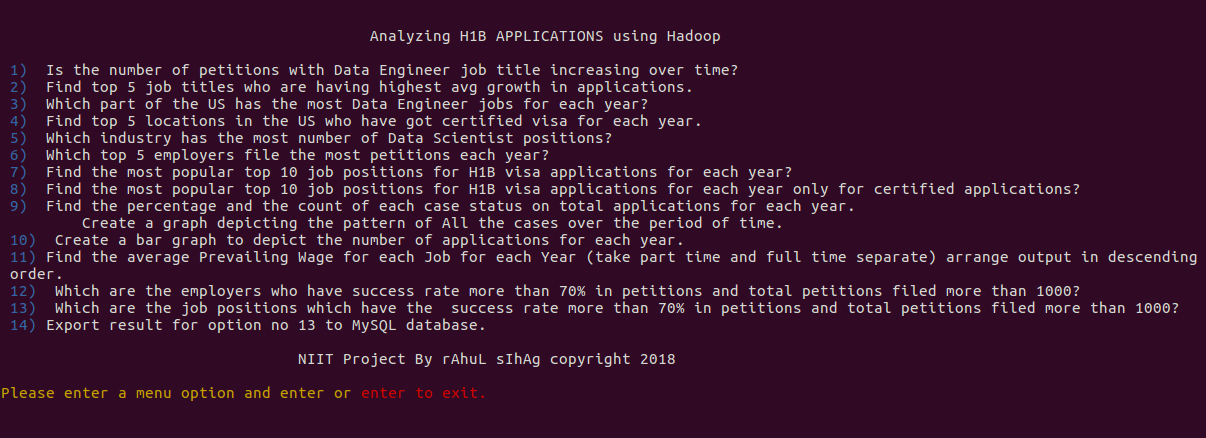
Step 3: look up/supporting tables to be created based on the analysis.

**MyMenu.sh**

Creating a menu for running the jobs stated in the project. Therefore queering over the data should be easy even for the person who does not have any knowledge about hadoop or codes.

It holds hive codes in itself so the queering should take least time for execution and results are shown ASAP.

rahul@apple:~$ chmod u+x MyMenu.sh

rahul@apple:~$ ./MyMenu.sh

Script for **MyMenu.sh** is as follows:

#!/bin/bash

show\_menu()

{

NORMAL=`echo "\033[m"`

MENU=`echo "\033[37m"` #Blue

NUMBER=`echo "\033[34m"` #yellow

FGRED=`echo "\033[41m"`

RED\_TEXT=`echo "\033[31m"`

ENTER\_LINE=`echo "\033[33m"`

echo -e "${MENU}\n\n\n\n\t\t\t\t\t Analyzing H1B APPLICATIONS using Hadoop\n ${NORMAL}"

echo -e "${MENU}${NUMBER} 1) ${MENU} Is the number of petitions with Data Engineer job title increasing over time?${NORMAL}"

echo -e "${MENU}${NUMBER} 2) ${MENU} Find top 5 job titles who are having highest growth in applications. ${NORMAL}"

echo -e "${MENU}${NUMBER} 3) ${MENU} Which part of the US has the most Data Engineer jobs for each year? ${NORMAL}"

echo -e "${MENU}${NUMBER} 4) ${MENU} Find top 5 locations in the US who have got certified visa for each year.${NORMAL}"

echo -e "${MENU}${NUMBER} 5) ${MENU} Which industry has the most number of Data Scientist positions?${NORMAL}"

echo -e "${MENU}${NUMBER} 6) ${MENU} Which top 5 employers file the most petitions each year? ${NORMAL}"

echo -e "${MENU}${NUMBER} 7) ${MENU} Find the most popular top 10 job positions for H1B visa applications for each year?${NORMAL}"

echo -e "${MENU}${NUMBER} 8) ${MENU} Find the most popular top 10 job positions for H1B visa applications for each year only for certified applications?${NORMAL}"

echo -e "${MENU}${NUMBER} 9) ${MENU} Find the percentage and the count of each case status on total applications for each year. \n\t Create a graph depicting the pattern of All the cases over the period of time.${NORMAL}"

echo -e "${MENU}${NUMBER} 10) ${MENU} Create a bar graph to depict the number of applications for each year.${NORMAL}"

echo -e "${MENU}${NUMBER} 11) ${MENU}Find the average Prevailing Wage for each Job for each Year (take part time and full time separate) arrange output in descending order.${NORMAL}"

echo -e "${MENU}${NUMBER} 12) ${MENU} Which are employers who have the highest success rate in petitions more than 70% in petitions and total petitions filed more than 1000?${NORMAL}"

echo -e "${MENU}${NUMBER} 13) ${MENU} Which are the job positions which have the success rate more than 70% in petitions and total petitions filed more than 1000? ${NORMAL}"

echo -e "${MENU}${NUMBER} 14) ${MENU}Export result for option no 11 to MySQL database.\n${NORMAL}"

echo -e "${MENU}\t\t\t\t NIIT Project By rAhuL sIhAg copyright 2018\n${NORMAL}"

echo -e "${ENTER\_LINE}Please enter a menu option and enter or ${RED\_TEXT}enter to exit. ${NORMAL}"

read opt

}

function option\_picked()

{

COLOR='\033[01;31m' # bold red

RESET='\033[00;00m' # normal white

MESSAGE="$1" #modified to post the correct option selected

echo -e "${COLOR}${MESSAGE}${RESET}"

}

clear

show\_menu

while [ opt != '' ]

do

if [[ $opt = "" ]]; then

exit;

else

case $opt in

1) option\_picked "1 a) Is the number of petitions with Data Engineer job title increasing over time?";

start-all.sh

hive -e " select year,count(year) as Count from visa.h1b\_final where job\_title like '%DATA ENGINEER%' group by year order by year asc;"

show\_menu;

;;

2) option\_picked "1 b) Find top 5 job titles who are having highest avg growth in applications. ";

stop-all.sh

pig -x local /home/rahul/project/Pig/question1b.pig

show\_menu;

;;

3) option\_picked "2 a) Which part of the US has the most Data Engineer jobs for each year?";

stop-all.sh

pig -x local /home/rahul/project/Pig/question2a.pig

show\_menu;

;;

4) option\_picked "2 b) find top 5 locations in the US who have got certified visa for each year.";

start-all.sh

echo -e "Enter the year (2011,2012,2013,2014,2015,2016)"

read var

hive -e " select year, worksite,count(case\_status) as allcase\_status from visa.h1b\_final where year =$var and case\_status='CERTIFIED' group by worksite,year order by allcase\_status desc limit 5;"

show\_menu;

;;

5) option\_picked "3) Which industry has the most number of Data Scientist positions?";

start-all.sh

hive -e " select soc\_name,count(soc\_name) as total\_data\_scientists from visa.h1b\_final where job\_title LIKE '%DATA SCIENTIST%' group by soc\_name order by total\_data\_scientists desc;"

show\_menu;

;;

6) option\_picked "4)Which top 5 employers file the most petitions each year?";

start-all.sh

hive -e "create view topemp as select employer\_name,year, count(case\_status) as petition\_filed from visa.h1b\_final where year in ('2011','2012','2013','2014','2015','2016') group by year, employer\_name sort by year, petition\_filed desc;

select year, employer\_name, petition\_filed ,rank from(select year, employer\_name, rank() over (partition by year order by petition\_filed desc) as rank,petition\_filed from topemp) ranked\_table where ranked\_table.rank <=5;"

show\_menu;

;;

7) option\_picked "5a) Find the most popular top 10 job positions for H1B visa applications for each year?";

start-all.sh

echo -e "Enter the year (2011,2012,2013,2014,2015,2016)"

read var

echo "Find the most popular top 10 job positions for H1B visa applications for each year?";

hive -e "select job\_title,year,count(case\_status ) as no\_of\_jobs from visa.h1b\_final where year= $var group by job\_title,year order by no\_of\_jobs desc limit 10; "

show\_menu;

;;

8) option\_picked "5b) Find the most popular top 10 job positions for H1B visa applications for each year only certified position?";

start-all.sh

echo -e "Enter the year (2011,2012,2013,2014,2015,2016)"

read var

echo "Find the most popular top 10 job positions for H1B visa applications for each year?";

hive -e "select job\_title,year,count(case\_status ) as no\_of\_jobs from visa.h1b\_final where year= $var and case\_status='CERTIFIED' group by job\_title,year order by no\_of\_jobs desc limit 10; "

show\_menu;

;;

9) stop-all.sh

option\_picked "6) Find the percentage and the count of each case status on total applications for each year. Create a graph depicting the pattern of all the cases over the period of time.";

pig -x local /home/rahul/project/Pig/question6.pig

show\_menu;

;;

10) start-all.sh

sleep 6

option\_picked "7) Create a bar graph to depict the number of applications for each year.";

hive -e "select year,count(\*) as applications from visa.h1b\_final where year like '201%' group by year;"

show\_menu;

;;

11) option\_picked "8) Find the average Prevailing Wage for each Job for each Year (take part time and full time separate) arrange output in descending order";

hive -f /home/rahul/project/Hive/question8.sql

show\_menu;

;;

12) option\_picked "9) Which are the employers along with the number of petitions who have the success rate more than 70% in petitions. (total petitions filed 1000 OR more than 1000)?"

stop-all.sh

pig -x local /home/rahul/project/Pig/question9.pig

show\_menu;

;;

13) option\_picked "10) Which are the job positions along with the number of petitions which have the success rate more than 70% in petitions (total petitions filed 1000 OR more than 1000)?"

stop-all.sh

rm -r /home/rahul/project/Pig/question10

pig -x local /home/rahul/project/Pig/question10.pig

cat /home/rahul/project/Pig/question10/p\*

show\_menu;

;;

14) option\_picked "11) Export result for question no 13 \*(Which are the job positions along with the number of petitions which have the success rate more than 70% in petitions (total petitions filed 1000 OR more than 1000)?)\* to MySql database."

start-all.sh

bash /home/rahul/project/Sqoop/question11.sh

show\_menu;

;;

\n) exit;

;;

\*) clear;

option\_picked "Pick an option from the menu";

show\_menu;

;;

esac

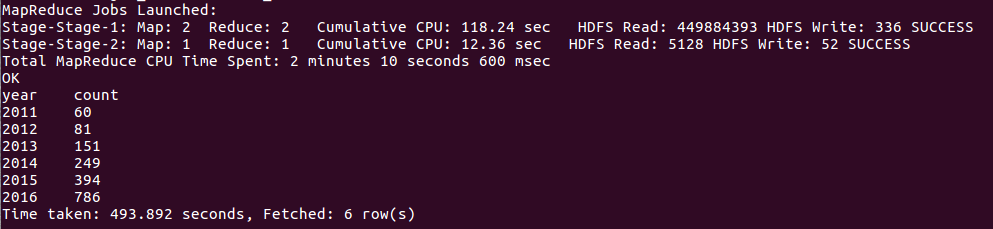
fi

done

**Job 1: Is the number of petitions with Data Engineer job title increasing over time?**

Technology used: **HIVE**

SELECT year, count(year) as count FROM visa.h1b\_final WHERE job\_title LIKE '%DATA ENGINEER%' GROUP BY year ORDER BY year ASC;

**Output:**

**year count**

2011 60

2012 81

2013 151

2014 249

2015 394

2016 786

Time taken: 493.892 seconds, Fetched: 6 row(s)

**Job 2: Find top 5 job titles who are having highest growth in applications.**

Technology used: **PIG**

--filtering dataset by year

a= group cleansed by $4; --grouping by job

step\_a= foreach a generate group,COUNT($1); --generate year,job,count

cleansed= filter noheader by $7 matches '2012'; --filtering dataset by year

a= group cleansed by $4; --grouping by job

step\_b= foreach a generate group,COUNT($1) --generate year,job,count

cleansed= filter noheader by $7 matches '2013'; --filtering dataset by year

a= group cleansed by $4; --grouping by job

step\_c= foreach a generate group,COUNT($1); --generate year,job,count

cleansed= filter noheader by $7 matches '2014'; --filtering dataset by year

a= group cleansed by $4; --grouping by job

step\_d= foreach a generate group,COUNT($1); --generate year,job,count

cleansed= filter noheader by $7 matches '2015'; --filtering dataset by year

a= group cleansed by $4; --grouping by job

step\_e= foreach a generate group,COUNT($1); --generate year,job,count

cleansed= filter noheader by $7 matches '2016'; --filtering dataset by year

a= group cleansed by $4; --grouping by job

step\_f= foreach a generate group,COUNT($1); --generate year,job,count

joined= join step\_a by $0,step\_b by $0,step\_c by $0,step\_d by $0,step\_e by $0,step\_f by $0;

yearwiseapplications= foreach joined generate $0,$1,$3,$5,$7,$9,$11; --generate progressive growth

progressivegrowth= foreach yearwiseapplications generate $0,

(float)($6-$5)\*100/$5,(float)($5-$4)\*100/$4,

(float)($4-$3)\*100/$3,(float)($3-$2)\*100/$2,

(float)($2-$1)\*100/$1; --average progressive growth

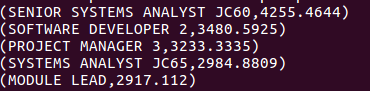
avgprogressivegrowth= foreach progressivegrowth generate $0,($1+$2+$3+$4+$5)/5; ---ordered progressive growth

orderedavggrowth= order avgprogressivegrowth by $1 desc; --display top5 only

answer = limit orderedavggrowth 5;

dump answer;

**Output:**



(SENIOR SYSTEMS ANALYST JC 60,4255.4644)

(SOFTWARE DEVELOPER 2,3480.5925)

(PROJECT MANAGER 3,3233.3335)

(SYSTEMS ANALYST JC65,2984.8809)

(MODULE LEAD,2917.112)

**Job 3: Which part of the US has the most Data Engineer jobs for each year?**

Technology used: **PIG**

REGISTER '/home/rahul/jar/piggybank.jar'; --Register external jar 'Piggy Bank.jar'

REGISTER '/home/rahul/jar/piggybank.jar'; --Register external jar 'Piggy Bank.jar'

DEFINE CSVExcelStorage org.apache.pig.piggybank.storage.CSVExcelStorage; -- within the jar define a function CSVExcelStorage()

data = LOAD '/home/rahul/project/h1b.csv' USING CSVExcelStorage() as

(s\_no:int,case\_status:chararray,employer\_name:chararray,soc\_name:chararray,job\_title:chararray,full\_time\_position:chararray,prevailing\_wage:int,year:chararray,worksite:chararray,longitute:double,latitute:double); --Load data

noheader= filter data by $0>=1; --Remove header

cleansed= filter noheader by $4 matches '.\*DATA ENGINEER.\*' and $7 matches '2011'; --filtering dataset which contains 'DATA ENGINEER' .

a= group cleansed by $8; --grouping by worksite

step\_a= foreach a generate '2011',group,COUNT($1); --generate year,worksite,count

step\_b= order step\_a by $2 desc;

answer\_2011= limit step\_b 1;

cleansed= filter noheader by $4 matches '.\*DATA ENGINEER.\*' and $7 matches '2012'; --filtering dataset which contains 'DATA ENGINEER' .

a= group cleansed by $8; --grouping by worksite

step\_a= foreach a generate '2012',group,COUNT($1); --generate year,worksite,count

step\_b= order step\_a by $2 desc;

answer\_2012= limit step\_b 1;

cleansed= filter noheader by $4 matches '.\*DATA ENGINEER.\*' and $7 matches '2013';

a= group cleansed by $8;

step\_a= foreach a generate '2013',group,COUNT($1);

step\_b= order step\_a by $2 desc;

answer\_2013= limit step\_b 1;

cleansed= filter noheader by $4 matches '.\*DATA ENGINEER.\*' and $7 matches '2014';

a= group cleansed by $8;

step\_a= foreach a generate '2014',group,COUNT($1);

step\_b= order step\_a by $2 desc;

answer\_2014= limit step\_b 1;

cleansed= filter noheader by $4 matches '.\*DATA ENGINEER.\*' and $7 matches '2015';

a= group cleansed by $8;

step\_a= foreach a generate '2015',group,COUNT($1);

step\_b= order step\_a by $2 desc;

answer\_2015= limit step\_b 1;

cleansed= filter noheader by $4 matches '.\*DATA ENGINEER.\*' and $7 matches '2016';

a= group cleansed by $8;

step\_a= foreach a generate '2016',group,COUNT($1);

step\_b= order step\_a by $2 desc;

answer\_2016= limit step\_b 1;

dump answer\_2011; --display answers for every year

dump answer\_2012;

dump answer\_2013;

dump answer\_2014;

dump answer\_2015;

dump answer\_2016;

**Output:**

SEATTLE, WASHINGTON 2011 20

SEATTLE, WASHINGTON 2012 30

SEATTLE, WASHINGTON 2013 46

SEATTLE, WASHINGTON 2014 45

SEATTLE, WASHINGTON 2015 61

SEATTLE, WASHINGTON 2016 128

**Job 4: Find top 5 locations in the US who have got certified visa for each year.**

Technology used: **HIVE**

echo -e "Enter the year (2011,2012,2013,2014,2015,2016)"

read var

hive -e " SELECT year, worksite,count(case\_status) as allcase\_status FROM visa.h1b\_final WHERE year =$var and case\_status='CERTIFIED' GROUP BY worksite,year ORDER BY allcase\_status DESC LIMIT 5;"

**Output:**

NEW YORK, NEW YORK 2011 23172

HOUSTON, TEXAS 2011 8184

CHICAGO, ILLINOIS 2011 5188

SAN JOSE, CALIFORNIA 2011 4713

SAN FRANCISCO, CALIFORNIA 2011 4711

NEW YORK, NEW YORK 2012 23737

HOUSTON, TEXAS 2012 9963

SAN FRANCISCO, CALIFORNIA 2012 6116

CHICAGO, ILLINOIS 2012 5671

ATLANTA, GEORGIA 2012 5565

NEW YORK, NEW YORK 2013 23537

HOUSTON, TEXAS 2013 11136

SAN FRANCISCO, CALIFORNIA 2013 7281

SAN JOSE, CALIFORNIA 2013 6722

ATLANTA, GEORGIA 2013 6377

NEW YORK, NEW YORK 2014 27634

HOUSTON, TEXAS 2014 13360

SAN FRANCISCO, CALIFORNIA 2014 9798

SAN JOSE, CALIFORNIA 2014 8223

ATLANTA, GEORGIA 2014 8213

NEW YORK, NEW YORK 2015 31266

HOUSTON, TEXAS 2015 15242

SAN FRANCISCO, CALIFORNIA 2015 12594

ATLANTA, GEORGIA 2015 10500

SAN JOSE, CALIFORNIA 2015 9589

NEW YORK, NEW YORK 2016 34639

SAN FRANCISCO, CALIFORNIA 2016 13836

HOUSTON, TEXAS 2016 13655

ATLANTA, GEORGIA 2016 11678

CHICAGO, ILLINOIS 2016 11064

**Job 5: Which industry has the most number of Data Scientist positions?**

Technology used: **HIVE**

SELECT soc\_name,count(soc\_name) as total\_data\_scientists FROM visa.h1b\_final WHERE job\_title LIKE '%DATA SCIENTIST%' GROUP BY soc\_name ORDER BY total\_data\_scientists DESC;

**Output Sample:**

**soc\_name** **total\_data\_scientists**

STATISTICIANS 649

COMPUTER AND INFORMATION RESEARCH SCIENTISTS 500

OPERATIONS RESEARCH ANALYSTS 426

COMPUTER AND INFORMATION RESEARCH SCIENTISTS 208

COMPUTER OCCUPATIONS, ALL OTHER 179

STATISTICIANS 152

SOFTWARE DEVELOPERS, APPLICATIONS 148

MATHEMATICIANS 147

COMPUTER SYSTEMS ANALYSTS 135

OPERATIONS RESEARCH ANALYSTS 124

SOFTWARE DEVELOPERS, SYSTEMS SOFTWARE 61

SOFTWARE DEVELOPERS, APPLICATIONS 59

COMPUTER OCCUPATIONS, ALL OTHER 45

COMPUTER SYSTEMS ANALYSTS 30

BIOLOGICAL SCIENTISTS, ALL OTHER 27

SOFTWARE DEVELOPERS, SYSTEMS SOFTWARE 22

DATABASE ADMINISTRATORS 17

FINANCIAL SPECIALISTS, ALL OTHER 15

MATHEMATICIANS 13

**Time taken: 401.659 seconds, Fetched: 87 row(s)**

**Job 6: Which top 5 employers file the most petitions each year?**

Technology used: **HIVE**

CREATE VIEW topemp as SELECT employer\_name,year, count(case\_status) as petition\_filed FROM visa.h1b\_final WHERE year in ('2011','2012','2013','2014','2015','2016') GROUP BY year, employer\_name SORT BY year, petition\_filed desc;

SELECT year, employer\_name, petition\_filed ,rank FROM(SELECT year, employer\_name, rank() over (PARTITION BY year ORDER BY petition\_filed DESC) as rank,petition\_filed FROM topemp) ranked\_table WHERE ranked\_table.rank <=5;

**year employer\_name petition\_filed rank**

2011 TATA CONSULTANCY SERVICES LIMITED 5416 1

2011 MICROSOFT CORPORATION 4253 2

2011 DELOITTE CONSULTING LLP 3621 3

2011 WIPRO LIMITED 3028 4

2011 COGNIZANT TECHNOLOGY SOLUTIONS U.S. CORPORATION 2721 5

2012 INFOSYS LIMITED 15818 1

2012 WIPRO LIMITED 7182 2

2012 TATA CONSULTANCY SERVICES LIMITED 6735 3

2012 DELOITTE CONSULTING LLP 4727 4

2012 IBM INDIA PRIVATE LIMITED 4074 5

2013 INFOSYS LIMITED 32223 1

2013 TATA CONSULTANCY SERVICES LIMITED 8790 2

2013 WIPRO LIMITED 6734 3

2013 DELOITTE CONSULTING LLP 6124 4

2013 ACCENTURE LLP 4994 5

2014 INFOSYS LIMITED 23759 1

2014 TATA CONSULTANCY SERVICES LIMITED 14098 2

2014 WIPRO LIMITED 8365 3

2014 DELOITTE CONSULTING LLP 7017 4

2014 ACCENTURE LLP 5498 5

2015 INFOSYS LIMITED 33245 1

2015 TATA CONSULTANCY SERVICES LIMITED 16553 2

2015 WIPRO LIMITED 12201 3

2015 IBM INDIA PRIVATE LIMITED 10693 4

2015 ACCENTURE LLP 9605 5

2016 INFOSYS LIMITED 25352 1

2016 CAPGEMINI AMERICA INC 16725 2

2016 TATA CONSULTANCY SERVICES LIMITED 13134 3

2016 WIPRO LIMITED 10607 4

2016 IBM INDIA PRIVATE LIMITED 9787 5

**Time taken: 1229.208 seconds, Fetched: 30 row(s)**

**Job 7: Find the most popular top 10 job positions for H1B visa applications for each year?**

Technology used: **HIVE**

echo -e "Enter the year (2011,2012,2013,2014,2015,2016)"

read var

echo "Find the most popular top 10 job positions for H1B visa applications for each year?";

hive -e "SELECT job\_title,year,count(case\_status ) as no\_of\_jobs FROM visa.h1b\_final WHERE year= $var GROUP BY job\_title,year ORDER BY no\_of\_jobs DESC LIMIT 10; "

**Output Sample:**

**job\_title year no\_of\_jobs**

PROGRAMMER ANALYST 2011 31799

SOFTWARE ENGINEER 2011 12763

COMPUTER PROGRAMMER 2011 8998

SYSTEMS ANALYST 2011 8644

BUSINESS ANALYST 2011 3891

COMPUTER SYSTEMS ANALYST 2011 3698

ASSISTANT PROFESSOR 2011 3467

PHYSICAL THERAPIST 2011 3377

SENIOR SOFTWARE ENGINEER 2011 2935

SENIOR CONSULTANT 2011 2798

**Time taken: 446.94 seconds, Fetched: 10 row(s)**

***Another way to execute the same query fetching all year’s data at once:***

*create view topjob as select job\_title,year, count(case\_status) as cnt from visa.h1b\_final where year in ('2011','2012','2013','2014','2015','2016') group by year, job\_title sort by year, cnt desc;*

*select year, job\_title, cnt ,rank from(select year, job\_title, rank() over (partition by year order by cnt desc) as rank,cnt from topjob) ranked\_table where ranked\_table.rank <=10;*

**Job 8: Find the most popular top 10 job positions for H1B visa applications for each year only for certified applications?**

Technology used: **HIVE**

SELECT job\_title,year,count(case\_status ) as sumstatus FROM visa.h1b\_final WHERE year= $var and case\_status='CERTIFIED' GROUP BY job\_title,year order by sumstatus DESC LIMIT 10;

**Output Sample:**

**job\_title year no\_of\_jobs**

PROGRAMMER ANALYST 2016 47964

SOFTWARE ENGINEER 2016 25890

SOFTWARE DEVELOPER 2016 12474

SYSTEMS ANALYST 2016 10986

COMPUTER PROGRAMMER 2016 10528

BUSINESS ANALYST 2016 8175

COMPUTER SYSTEMS ANALYST 2016 6205

DEVELOPER 2016 5912

SENIOR SOFTWARE ENGINEER 2016 5630

TECHNOLOGY LEAD - US 2016 5405

**Time taken: 745.107 seconds, Fetched: 10 row(s)**

**Job 9: Find the percentage and the count of each case status on total applications for each year. Create a graph depicting the pattern of all the cases over the period of time.**

Technology used: **PIG**

REGISTER '/home/rahul/jar/piggybank.jar';--Register external jar 'Piggy Bank.jar'

REGISTER '/home/rahul/jar/piggybank.jar';--Register external jar 'Piggy Bank.jar'

DEFINE CSVExcelStorage org.apache.pig.piggybank.storage.CSVExcelStorage; -- within the jar define a function CSVExcelStorage()

data = LOAD '/home/rahul/project/h1b.csv' USING CSVExcelStorage() as

(s\_no:int,case\_status:chararray,employer\_name:chararray,soc\_name:chararray,job\_title:chararray,full\_time\_position:chararray,prevailing\_wage:int,year:chararray,worksite:chararray,longitute:double,latitute:double); --load data

noheader= filter data by $0>=1; --remove header

cleansed= filter noheader by $1 is not null and $1!='NA';

temp= group cleansed by $7;

total= foreach temp generate group,COUNT(cleansed.$1); --describe total;dump total;

noheader= filter data by $0>=1;

cleansed= filter noheader by $7 is not null and $7!='NA';

temp= group cleansed by ($7,$1);

yearsoccount= foreach temp generate group,group.$0,COUNT($1);

dump yearsoccount;

joined= join yearsoccount by $1,total by $0;

ans= foreach joined generate FLATTEN($0),(float)($2\*100)/$4,$2; --percent generation

dump ans;

**OUTPUT:**

(2011,DENIED,8.119476,29130)

(2011,CERTIFIED-WITHDRAWN,3.2321813,11596)

(2011,WITHDRAWN,2.8165913,10105)

(2011,CERTIFIED,85.83175,307936)

(2012,WITHDRAWN,2.5805628,10725)

(2012,CERTIFIED-WITHDRAWN,7.487362,31118)

(2012,DENIED,5.075949,21096)

(2012,CERTIFIED,84.856125,352668)

(2013,WITHDRAWN,2.6214957,11590)

(2013,CERTIFIED-WITHDRAWN,8.014222,35432)

(2013,DENIED,2.7427316,12126)

(2013,CERTIFIED,86.61816,382951)

(2013,PENDING QUALITY AND COMPLIANCE REVIEW - UNASSIGNED, 0.00339279,15)

(2014,DENIED,2.290216,11896)

(2014,CERTIFIED-WITHDRAWN,6.998096,36350)

(2014,INVALIDATED,1.9251983E-4,1)

(2014,WITHDRAWN,3.086863,16034)

(2014,CERTIFIED,87.624245,455144)

(2014,REJECTED,3.8503966E-4,2)

(2015,DENIED,1.765399,10923)

(2015,CERTIFIED,88.452255,547278)

(2015,WITHDRAWN,3.1443594,19455)

(2015,CERTIFIED-WITHDRAWN,6.6379843,41071)

(2016,DENIED,1.4163257,9175)

(2016,CERTIFIED,87.93507,569646)

(2016,WITHDRAWN,3.3791137,21890)

(2016,CERTIFIED-WITHDRAWN,7.269494,47092)

**Job 10: Create a bar graph to depict the number of applications for each year.**

Technology used: **HIVE**

SELECT year, count(\*) applications FROM h1b\_final GROUP BY year ORDER BY year;

**OUTPUT:**

2011 358767

2012 415607

2013 442114

2014 519427

2015 618727

2016 647803

H1B Applications

**Job 11: Find the average Prevailing Wage for each Job for each Year (take part time and full time separate) arrange output in descending order.**

Technlogy use: **PIG**

finding the avg prevailing wage for each job per year, finding the counting the number of full time position and finding the sum of prevailing wage.

**Pig code:**

**Pig code for the year 2011:**

a = LOAD '/home/rahul/project/h1b.csv' USING PigStorage() as (s\_no:int,case\_status:chararray,employer\_name:chararray,soc\_name:chararray,job\_title:chararray,full\_time\_position:chararray,prevailing\_wage:int,year:chararray,worksite:chararray,longitute:double,latitute:double);

b = filter a by year=='2011';

c = foreach b generate $4, $5, $6, $7;

d = group c by ($0, $1);

e = foreach d generate group as job\_title, COUNT(c),SUM(c.prevailining\_wage);

f = foreach e generate $0, ($2/$1)as avg;

g = order f by $0 desc;

h = limit g 10;

dump h;

**sample output:**

2011

(( SYSTEMS ANALYST,Y),42078.5)

(( COMPUTER SYSTEMS ENGINEER,Y),46218.0)

((|NFORMATION MANAGEMENT SPECIALIST,Y),38875.0)

(([PHYSICAL THERAPIST,Y),69035.0)

(([HIOX] COMMERCIAL SPECIALIST (SALES ENGINEER),Y),63357.0)

((ZOOLOGIST,Y),40914.0)

((ZONE MANAGER, OPERATIONS & ANALYSIS,Y),89378.0)

((ZONE BUSINESS DEVELOPMENT MANAGER,Y),80912.0)

((YOUTUBE STRATEGY & OPERATIONS ANALYST,Y),72238.0)

((YOUTH SERVICE DEPARTMENT SUPERVISOR,Y),30514.0)

**Pig code for the year 2012:**

a = LOAD '/home/rahul/project/h1b.csv' USING PigStorage() as (s\_no:int,case\_status:chararray,employer\_name:chararray,soc\_name:chararray,job\_title:chararray,full\_time\_position:chararray,prevailing\_wage:int,year:chararray,worksite:chararray,longitute:double,latitute:double);

b = filter a by year=='2012';

c = foreach b generate $4, $5, $6, $7;

d = group c by ($0, $1);

e = foreach d generate group as job\_title, COUNT(c),SUM(c.prevailining\_wage);

f = foreach e generate $0, ($2/$1)as avg;

g = order f by $0 desc;

h = limit g 10;

dump h;

**Sample Output:**

(( LEAD TEST ANALYST,Y),69389.0)

((ZOOLOGISTS AND WILDLIFE BIOLOGISTS I,Y),31803.0)

((ZOOLOGIST - REPRODUCTIVE PHYSIOLOGY,Y),56222.0)

((ZOOKEEPER,Y),20800.0)

((ZOO BIRD KEEPER,Y),30784.0)

((ZONING MANAGER,Y),84635.0)

((ZONE MERCHANDISER,Y),64064.0)

((ZLC SPECIALIST, PROCESS IMPROVEMENT,Y),67080.0)

((YOUTH WORKER,Y),24044.0)

((YOUTH THERAPIST,Y),28558.0)

**Pig code for the year 2013:**

a = LOAD '/home/rahul/project/h1b.csv' USING PigStorage() as (s\_no:int,case\_status:chararray,employer\_name:chararray,soc\_name:chararray,job\_title:chararray,full\_time\_position:chararray,prevailing\_wage:int,year:chararray,worksite:chararray,longitute:double,latitute:double);

b = filter a by year=='2013';

c = foreach b generate $4, $5, $6, $7;

d = group c by ($0, $1);

e = foreach d generate group as job\_title, COUNT(c),SUM(c.prevailining\_wage);

f = foreach e generate $0, ($2/$1)as avg;

g = order f by $0 desc;

h = limit g 10;

dump h;

**Sample Output:**

(( TEST ANALYST - US,Y),53872.0)

(( TECHNOLOGY ARCHITECT - US,Y),96033.0)

(( LEAD CONSULTANT - US,Y),99652.0)

(( CONSULTANT - US,Y),56992.0)

(([FINANCIAL] ANALYST, STRUCTURED CREDIT,Y),73070.0)

((ZYQAD SPECIALIST,Y),88493.5)

((ZOO EDUCATION COORDINATOR,Y),36899.0)

((ZONE ACCOUNT MANAGER, CAPITOL DEAL,Y),30017.0)

((ZMS SOFTWARE ENGINEER,Y),105206.0)

((ZIMBABWE PARTNERSHIP COORDINATOR,N),25729.0)

**Pig code for the year 2014**

a = LOAD '/home/rahul/project/h1b.csv' USING PigStorage() as (s\_no:int,case\_status:chararray,employer\_name:chararray,soc\_name:chararray,job\_title:chararray,full\_time\_position:chararray,prevailing\_wage:int,year:chararray,worksite:chararray,longitute:double,latitute:double);

b = filter a by year=='2014';

c = foreach b generate $4, $5, $6, $7;

d = group c by ($0, $1);

e = foreach d generate group as job\_title, COUNT(c),SUM(c.prevailining\_wage);

f = foreach e generate $0, ($2/$1)as avg;

g = order f by $0 desc;

h = limit g 10;

dump h;

: **Sample Output:**

(( TEAM LEAD - US,Y),46779.0)

(( SOFTWARE TEST ENGINEER,Y),65936.0)

(( SENIOR PROJECT LEADER,Y),60778.0)

(( QUALITY ASSURANCE ANALYST,Y),77938.0)

(( MOBILE SQA ENGINEER   ,Y),41288.0)

(( BUSINESS INTELLIGENCE ANALYST,Y),77938.0)

((`QUALITY ASSURANCE ANALYST,Y),55682.0)

((]ENGINEERING LEAD,Y),51542.0)

((ZOOLOGIST,N),49795.0)

((ZOOKEEPER,Y),27745.0)

**Pig code for the year 2015**

a = LOAD '/home/rahul/project/h1b.csv' USING PigStorage() as (s\_no:int,case\_status:chararray,employer\_name:chararray,soc\_name:chararray,job\_title:chararray,full\_time\_position:chararray,prevailing\_wage:int,year:chararray,worksite:chararray,longitute:double,latitute:double);

b = filter a by year=='2015';

c = foreach b generate $4, $5, $6, $7;

d = group c by ($0, $1);

e = foreach d generate group as job\_title, COUNT(c),SUM(c.prevailining\_wage);

f = foreach e generate $0, ($2/$1)as avg;

g = order f by $0 desc;

h = limit g 10;

dump h;

**Sample Output:**

(( SYSTEMS ANALYST,Y),61776.0)

(( SOFTWARE TEST ENGINEER,Y),78707.0)

(( SAS ANALYST,Y),55598.0)

(( ORACLE APPS DBA,Y),60674.0)

((  MIDDLEWARE ADMINISTRATION.,Y),57429.0)

((ZOS SYSTEMS PROGRAMMER,Y),87818.0)

((ZONE ENGINEER,Y),82118.0)

((ZONAL ISOLATION SEGMENT ENGINEERING TECHNICAL AUTHORITY,Y),112486.0)

((ZMS WEB CLIENT ENGINEER,Y),108534.0)

((ZMS ENGINEER,Y),89502.0)

**Pig code for the year 2016**

a = LOAD '/home/rahul/project/h1b.csv' USING PigStorage() as (s\_no:int,case\_status:chararray,employer\_name:chararray,soc\_name:chararray,job\_title:chararray,full\_time\_position:chararray,prevailing\_wage:int,year:chararray,worksite:chararray,longitute:double,latitute:double);

b = filter a by year=='2016';

c = foreach b generate $4, $5, $6, $7;

d = group c by ($0, $1);

e = foreach d generate group as job\_title, COUNT(c),SUM(c.prevailining\_wage);

f = foreach e generate $0, ($2/$1)as avg;

g = order f by $0 desc;

h = limit g 10;

dump h;

**Sample Output:**

(( SR. BUSINESS INTELLIGENCE DEVELOPER,N),69909.0)

(( SOFTWARE PROGRAMMER,N),65042.0)

(( SOFTWARE ENGINEER,N),65042.0)

(( SHAREPOINT/SQL DEVELOPER,Y),76107.0)

(( QA ANALYST,N),56555.0)

(( PROJECT MANAGERS,N),60986.0)

(( BUSINESS INTELLIGENCE ANALYST,Y),77230.0)

(( BUSINESS ANALYST,N),60133.0)

((  MIDDLEWARE ADMINISTRATION.,N),57429.0)

(([FINANCIAL] ANALYST, STRUCTURED CREDIT,Y),80163.0)

**Job 12: Which are employers who have the highest success rate in petitions more than 70% in petitions and total petitions filed more than 1000?**

Finding the number of petitions and case\_status certified and certified-withdrawn for each employer\_name, then adding the number of these both case\_status then finding the success rate for each employer name, then checking the condition like success rate >70 in petitions filed more than 1000.

Technology used: **PIG**

REGISTER '/home/rahul/jar/piggybank.jar';--Register external jar 'Piggy Bank.jar'

REGISTER '/home/rahul/jar/piggybank.jar';--Register external jar 'Piggy Bank.jar'

DEFINE CSVExcelStorage org.apache.pig.piggybank.storage.CSVExcelStorage; -- within the jar define a function CSVExcelStorage()

data = LOAD '/home/rahul/project/h1b.csv' USING CSVExcelStorage() as (s\_no:int, case\_status:chararray, employer\_name:chararray, soc\_name:chararray, job\_title:chararray, full\_time\_position:chararray, prevailing\_wage:int, year:chararray, worksite:chararray, longitute:double, latitute:double);

noheader= filter data by $0>=1; --Remove header

cleansed= filter noheader by $1 is not null and $1!='NA';

temp= group cleansed by $2;

total= foreach temp generate group,COUNT(cleansed.$1); --Group by employername and count the case statuses as a whole

certified= filter noheader by $1 == 'CERTIFIED';

temp1= group certified by $2;

totalcertified= foreach temp1 generate group,COUNT(certified.$1); --Group by employername and count the case status = 'CERTIFIED'

certified= filter noheader by $1 == 'CERTIFIED-WITHDRAWN';

temp2= group certified by $2;

totalcertifiedwithdrawn= foreach temp2 generate group,COUNT(certified.$1); --Group by employername and count the case status = 'CERTIFIED-WITHDRAWN'

joined= join totalcertified by $0,totalcertifiedwithdrawn by $0,total by $0;

--dump joined;

joined= foreach joined generate $0,$1,$3,$5;

intermediateoutput= foreach joined generate $0,(float)($1+$2)\*100/($3),$3;

intermediateoutput2= filter intermediateoutput by $1>70 and $2>1000; --Filter by success-rate greater than 70% and petition count above 1000

finaloutput= order intermediateoutput2 by $1 DESC;

dump finaloutput;

**Sample Output:**

(INFOSYS LIMITED,99.54055,130592)

(ACCENTURE LLP,99.39307,33447)

(TATA CONSULTANCY SERVICES LIMITED,99.337204,64726)

(HCL AMERICA, INC.,99.26801,22678)

(RELIABLE SOFTWARE RESOURCES, INC.,99.14658,1992)

(NTT DATA, INC.,99.13251,4611)

(ERP ANALYSTS, INC.,99.10364,1785)

(PATNI AMERICAS INC.,99.07907,3149)

(KFORCE INC.,99.06015,1596)

(GENPACT LLC,98.852776,1046)

(SMARTPLAY, INC.,98.83805,1377)

(SYNTEL CONSULTING INC.,98.8317,3167)

(CREDIT SUISSE SECURITIES (USA) LLC,98.82168,2546)

(MASTECH, INC., A MASTECH HOLDINGS, INC. COMPANY,98.81408,5228)

(GENESIS ELDERCARE REHABILITATION SERVICES, INC.,98.78788,1320)

(HORIZON TECHNOLOGIES INC,98.78683,1731)

(SYNTEL INC,98.7667,1946)

(THE BOSTON CONSULTING GROUP, INC.,98.74261,1352)

(AMDOCS INC.,98.729225,1023)

(SAP AMERICA, INC.,98.69505,1456)

(DELOITTE TAX LLP,98.64054,2501)

(MPHASIS CORPORATION,98.63435,5199)

(3I INFOTECH, INC.,98.579124,2041)

(COMPUNNEL SOFTWARE GROUP, INC.,98.57904,3378)

(THE MATHWORKS, INC.,98.46535,2020)

(PERFICIENT, INC.,98.46266,1366)

(DALLAS INDEPENDENT SCHOOL DISTRICT,98.4384,1729)

(CGI TECHNOLOGIES AND SOLUTIONS INC.,98.39599,1995)

(VEDICSOFT,98.37468,1169)

(UNIVERSITY OF PITTSBURGH,98.34559,1632)

(DELOITTE CONSULTING LLP,98.32889,36742)

(BLOOMBERG, LP,98.29932,2352)

**Job 13: Which are the positions which have the success rate more than 70% in petitions and total petitions filed more than 1000?**

Finding the number of petitions and case\_status certified and certified-withdrawn for each employer\_name, then adding the number of these both case\_status then finding the success rate for each employer name, then checking the condition like success rate >70 in petitions filed more than 1000.

Technology used: **PIG**

--10) Which are the top 10 job positions that have success rate more than 70% in petitions and total petitions filed more than 1000?

REGISTER '/home/rahul/jar/piggybank.jar';--Register external jar 'Piggy Bank.jar'

REGISTER '/home/rahul/jar/piggybank.jar';--Register external jar 'Piggy Bank.jar'

DEFINE CSVExcelStorage org.apache.pig.piggybank.storage.CSVExcelStorage; -- within the jar define a function CSVExcelStorage()

data = LOAD '/home/rahul/project/h1b.csv' USING CSVExcelStorage() as (s\_no:int, case\_status:chararray, employer\_name:chararray, soc\_name:chararray, job\_title:chararray, full\_time\_position:chararray, prevailing\_wage:int, year:chararray, worksite:chararray, longitute:double, latitute:double);

noheader= filter data by $0>=1; --Remove header --Count Total Applications

cleansed= filter noheader by $1 is not null and $1!='NA';

temp= group cleansed by $4;

total= foreach temp generate group,COUNT(cleansed.$1); --Count Total Applications who are 'CERTIFIED'

certified= filter noheader by $1 == 'CERTIFIED';

temp1= group certified by $4;

totalcertified= foreach temp1 generate group,COUNT(certified.$1);

--Count Total Applications who are 'CERTIFIED-WITHDRAWN'

certified= filter noheader by $1 == 'CERTIFIED-WITHDRAWN';

temp2= group certified by $4;

totalcertifiedwithdrawn= foreach temp2 generate group,COUNT(certified.$1);

--SUCCESS\_RATE=(CERTIFIED+CERTIFIED-WITHDRAWN)/TOTAL X 100

joined= join totalcertified by $0,totalcertifiedwithdrawn by $0,total by $0;

joined= foreach joined generate $0,$1,$3,$5;

intermediateoutput= foreach joined generate $0,(float)($1+$2)\*100/($3),$3;

intermediateoutput2= filter intermediateoutput by $1>70 and $2>1000; --Filter by success-rate greater than 70% and petition count above 1000

finaloutput= order intermediateoutput2 by $1 DESC;

--STORE DATA INTO TEXT FILE

store finaloutput into '/home/rahul/project/Pig/question10' using PigStorage('\t');

**Sample Output:**

COMPUTER PROGRAMMER / CONFIGURER 2 100.0 1276

ASSOCIATE CONSULTANT - US 99.93171 4393

SYSTEMS ENGINEER - US 99.90036 10036

TEST ANALYST - US 99.818474 4958

CONSULTANT - US 99.81147 7426

TECHNOLOGY LEAD - US 99.80247 28350

TECHNICAL TEST LEAD - US 99.79531 5374

TECHNOLOGY ARCHITECT - US 99.766304 4707

TECHNOLOGY ANALYST - US 99.76204 26055

SENIOR PROJECT MANAGER - US 99.74766 2774

DEVELOPER USER INTERFACE 99.71412 5247

COMPUTER SYSTEMS ANALYST 2 99.70231 4031

SYSTEMS ANALYST - II 99.70127 1339

PROJECT MANAGER - III 99.69715 1651

PROJECT MANAGER - US 99.68777 7046

PROGRAMMER ANALYST - II 99.66555 3588

**Job 14: Export result for option no 12 to MySQL database.**

Technology used: **Sqoop**

#11) Export result for question no 10 \*(Which are the job positions along with the number of petitions which have the success rate more than 70% in petitions (total petitions filed 1000 OR more than 1000)?)\* to MySql database.

#Create a Database in mysql and create a table in it

hadoop fs -rm -r -f /Pig/Question10

hadoop fs -mkdir -p /Pig/Question10

hadoop fs -put /home/rahul/project/Pig/question10/p\* /Pig/Question10/

mysql -u root -p -e 'create database if not exists question11;use question11;create table question11(job\_title varchar(100),success\_rate float,petitions int);';

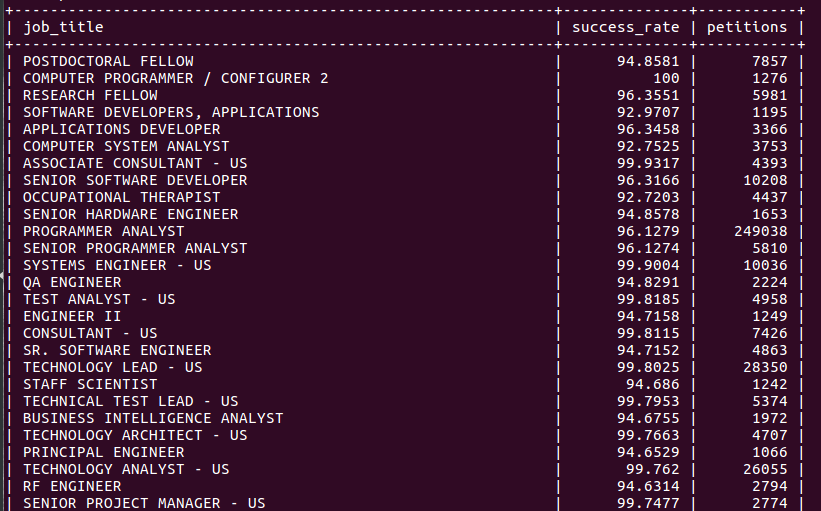
sqoop export --connect jdbc:mysql://localhost/question11 --username root --password 'macbook' --table question11 --export-dir /Pig/Question10/ --input-fields-terminated-by '\t' ;

echo -e '\n\nDisplay contents from MySQL Database.\n\n'

echo -e '\n10) Which are the top job positions that have success rate more than 70% in petitions and total petitions filed more than 1000?\n\n'

mysql -u root -p -e 'select \* from question11.question11';

**Sample Output:**



**Conclusion**

Following conclusions can draw based on task performed:-

1. MapReduce code in java makes the complex analysis quite easy. Code Required to be written to collect user input and performing complex join operations are handled efficiently by using this approach.
2. Hive helps in the cleaning up of the data. csv data can be easily converted into text format using hive.
3. For normal group by join and filter based data retrieval, pig is very efficient
4. Sqoop is useful when we have data in sql tables that needs to be imported in the hadoop file system.

**Reference**

* <http://www.student.niitcloudcampus.com/CloudCampus/Welcome.aspx>
* <https://www.sas.com/en_us/insights/big-data/hadoop.html>
* <https://www.google.co.in>