**Technical Report with Detailed Code Appendix**

**Patent Data Extraction**

**- Phase 1**

**Members:**

**BABU SATHYA SAI VAKKALAGADDA**

**AARADHANA RAGHUNATHAN**

**SAILESH BHAMIDIPATI**

**SHIVA NAGA PRASAD GUNDA**

**PAVAN KUMAR KANDULA**

**CHRISTOPHER LIS**

**PRANATHI LETHAKULA**

**TECHNICAL SUMMARY**

**PATENTS EXTRACTION:**

***Tools used: Hadoop-Hive, Schell Script.***

Following are the steps involved in Data Extraction:

1. To use a Schell script to extract Fortune 500 patents from the Hadoop cluster.
2. Run a Hive code on the extracted data to extract the desired columns.
3. Copy the contents to a single spreadsheet.

**Step 1:** To extract Fortune500 patents from cluster:

Below is the Schell script Code we used on to get Patent data from Zipped filed

#!/in/sh

 find -L ../patent\_data |grep -i ".zip" >Biblio\_files\_info.txt

 path=""

lines=$(cat Biblio\_files\_info.txt|wc -l)

for (( a=1; a<=$lines;a++ ))

do

path=$(sed -n "$a"p Biblio\_files\_info.txt )

unzip -u $path -d Biblio\_unzip\_files/

done

awk -v RS="</us-patent" '/<orgname>/{print}' Biblio\_unzip\_files/ipgb\*.xml > Patent\_data.txt

* This script extracts the Patents by unzipping the ipgb files from /projects/class/dsba-6100/patentData2000\_2015/patBiblio2000\_1h2015 folder in the cluster.
* AWK command used in the script “awk -v RS="</us-patent" '/<orgname>/{print}' Biblio\_unzip\_files / ipgb\*.xml > Patent\_data.txt” extracts patent data.

**Step 2:**

Ran Hive code on the extracted data to extract the desired columns. Once we have the Output file (obtained by running the script in particular because of the AWK command)

***CODE:***

Copying Extracted File to Hadoop Cluster:

hadoop fs -put *Patent\_data.txt* /user/sgunda1/

Creating Table and loading Patent\_data.txt:

create table all\_patents (file string);

LOAD DATA INPATH '/user/sgunda1/ *Patent\_data.txt* ' OVERWRITE INTO TABLE all\_patents;

Copying <orgname> rows to new table:

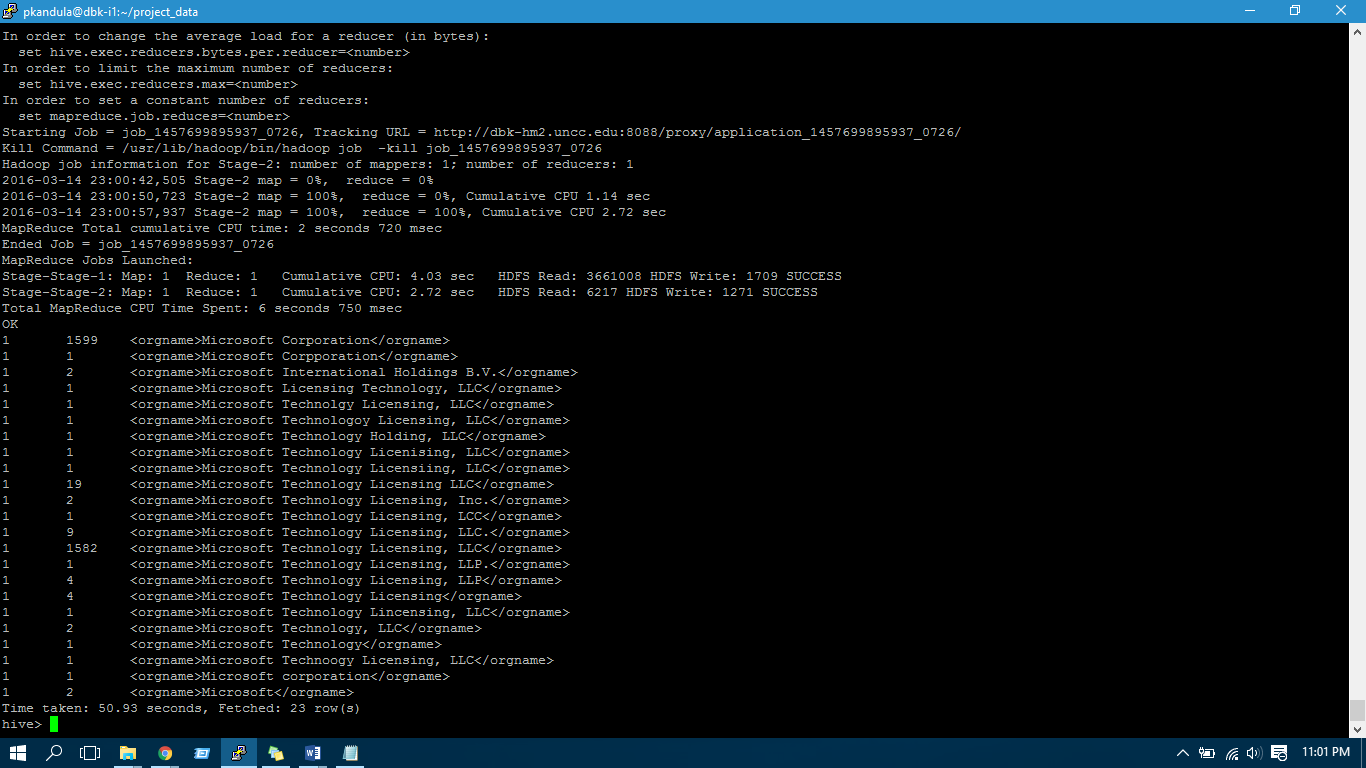
CREATE TABLE final\_patents\_count AS SELECT count(\*) AS count, file FROM all\_patents where file like '<orgname>%</orgname>' GROUP BY file ORDER BY file;

Copying Table to CSV File:

insert overwrite directory '/user/sgunda1/patents\_table' select \* FROM final\_patents\_count;

Getting Count for Microsoft:

select count(\*), SUM(count), file from final\_patents\_count where file like '<orgname>Microsoft%' group by file order by file;

******

**Top 50 Patent Terms Extraction:**

Following are the steps involved in Top 50 Patent Terms Extraction:

1. To use a Schell script to extract patents related to Specific Company.
2. Run a Hive code on the extracted data to extract the desired columns.
3. Using RStudio to get the Top 50 Words from the extracted data from Hive.

**Step1: (Continuation from Step1 in Above Patents Extraction Part)**

Below Code is used to copy Microsoft Patents data to each\_company\_data.txt File:

awk -v RS="</us-patent" '/<orgname>Microsoft/{print}' Biblio\_unzip\_files/ipgb\*.xml > each\_company\_data.txt;

Copying File to Hadoop Cluster:

hadoop fs -put each\_company\_data.txt /user/sgunda1/;

**Step2: (Hive code to extract Abstract column from Patent Info)**

Creating table to load patent data:

CREATE TABLE each\_company\_patents (file STRING);

Loading data to Table:

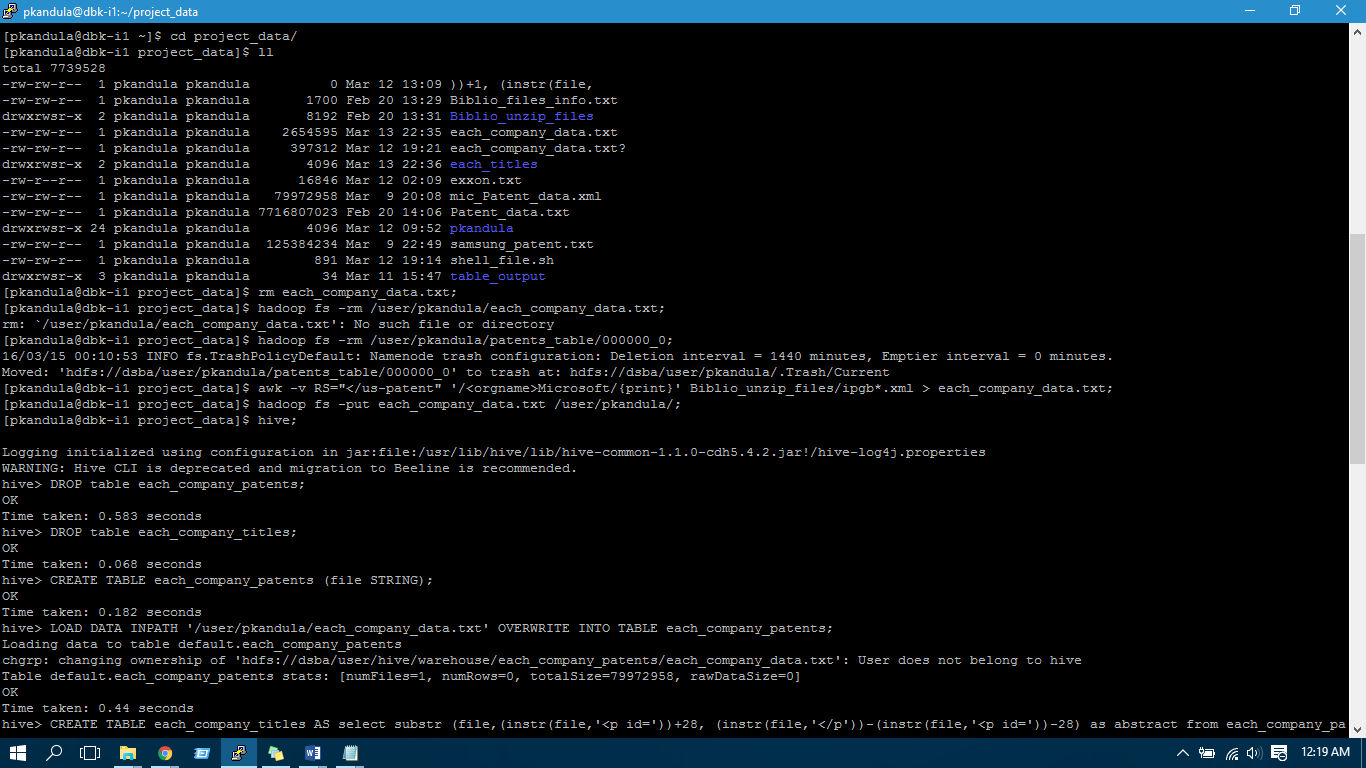
LOAD DATA INPATH '/user/sgunda1/each\_company\_data.txt' OVERWRITE INTO TABLE each\_company\_patents;

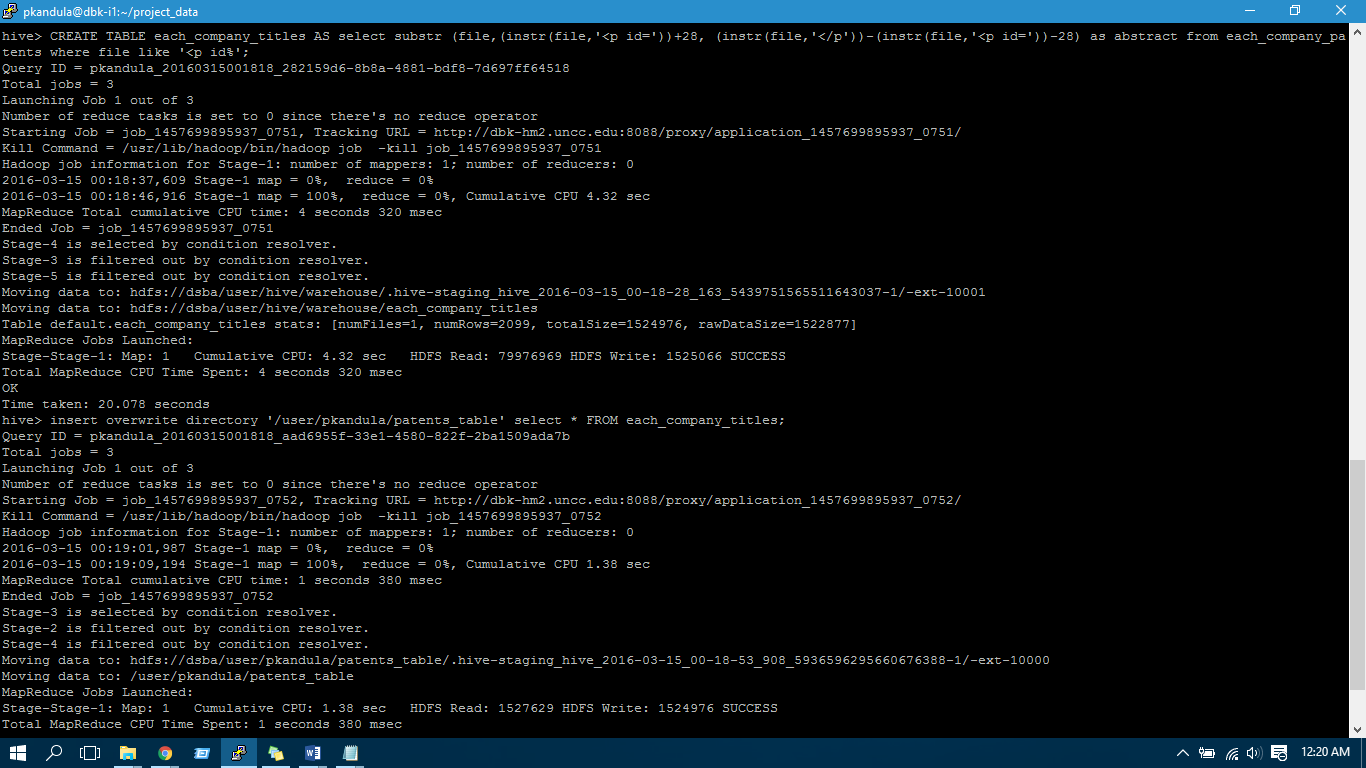
Creating new table to copy abstract column:

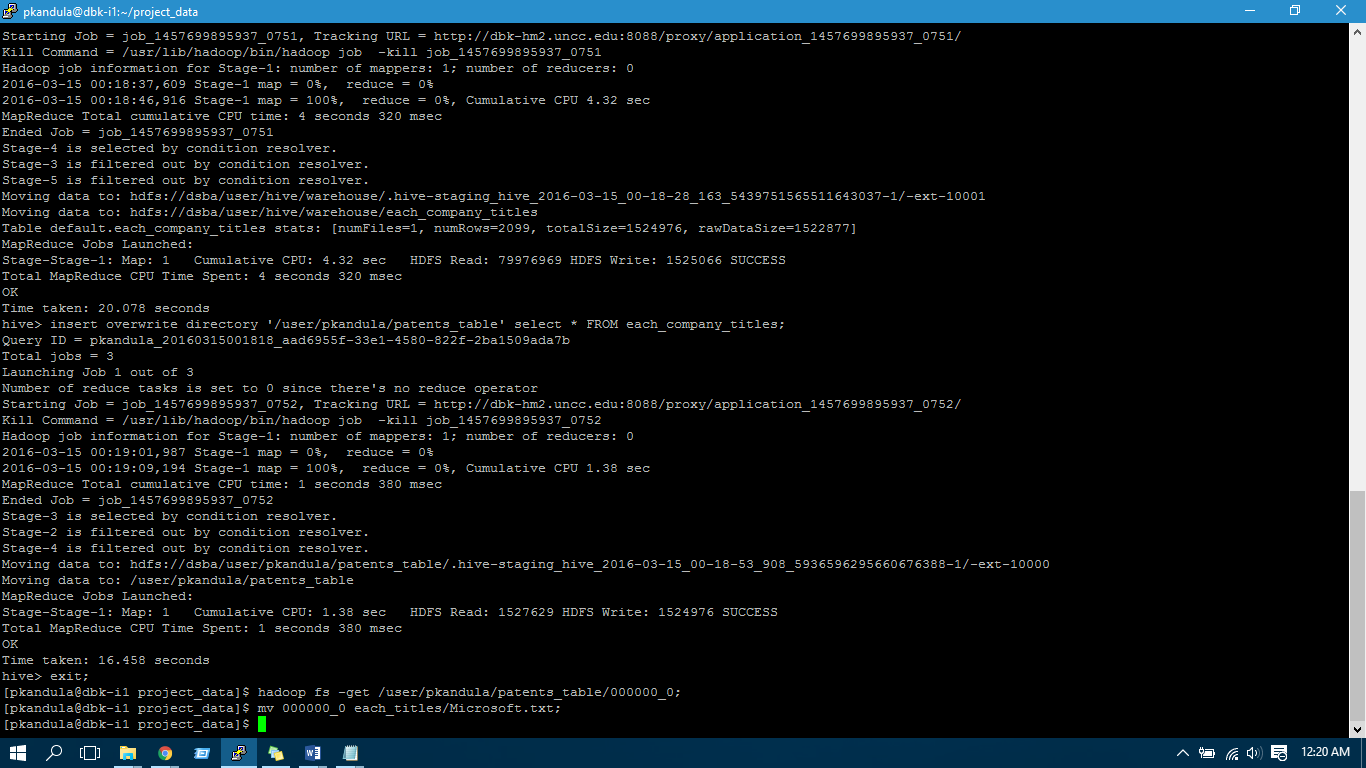
CREATE TABLE each\_company\_titles AS select substr (file,(instr(file,'<p id='))+28, (instr(file,'</p'))-(instr(file,'<p id='))-28) as abstract from each\_company\_patents where file like '<p id%';

Copying table to file:

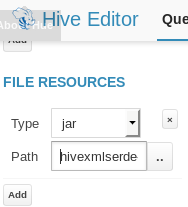
insert overwrite directory '/user/sgunda1/patents\_table' select \* FROM each\_company\_titles;







After Successful installation of the above software’s. We need to place the jar file **hivexmlserde-1.0.5.3.jar** in Hive environment. This class is used to work with XML files.



***Placing the Jar file in Hive Editor****.*

***CODE:***

***Creating Table to load XML data:***

*create table patent\_ipgb*

*(file string)*

*ROW FORMAT SERDE 'com.ibm.spss.hive.serde2.xml.XmlSerDe'*

*STORED AS*

*INPUTFORMAT 'com.ibm.spss.hive.serde2.xml.XmlSerDe'*

*OUTPUTFORMAT 'org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat'*

*TBLPROPERTIES (*

*"xmlinput.start"="<us-patent-grant",*

*"xmloutput.end"="</ us-patent-grant >"*

*);*

-------------------------------------------------------------------------------------------------------------------------------------*load data inpath 'hdfs:/user/hive/ipgb.xml' into TABLE patent\_ipgb* --------------------------------------------------------------------------------------------------------------------------------------

***Text parsing to get patent\_no, date-publ, date-produced:***

*select substr(*

*file,*

*(instr(file,'le="'))+4,*

*(instr(file,'.XML'))-(instr(file,'file='))-6)*

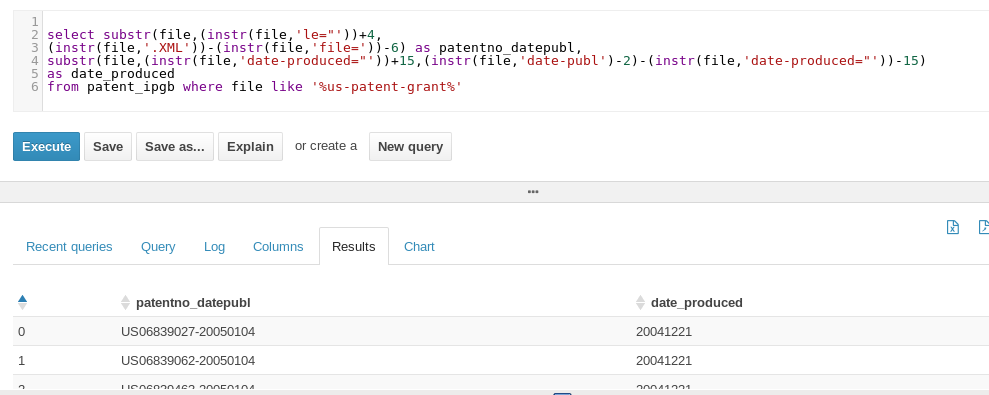
*as patentno\_datepubl,*

*substr(file,(instr(file,'date-produced="'))+15,*

*(instr(file,'date-publ')-2)-(instr(file,'date-produced="'))-15)*

*as date\_produced*

*from patent\_ipgb where file like '%us-patent-grant%'*



***patentno\_datepubl, date\_produced fields are extracted using Hive.***

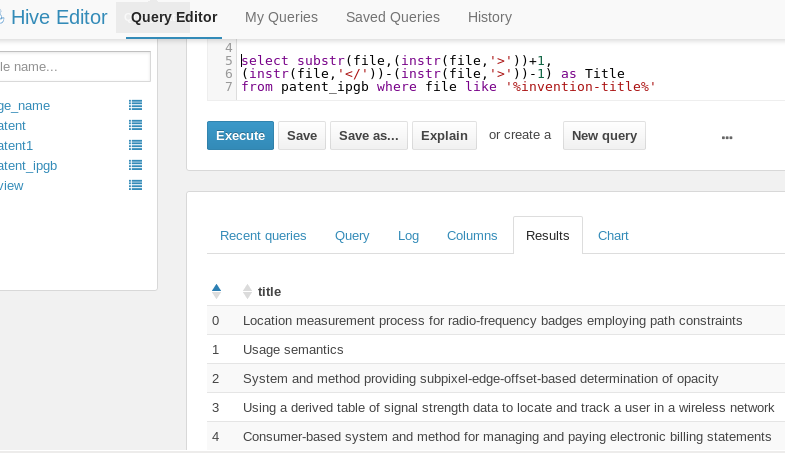
***Text parsing to get invention-title:***

*select substr(file,(instr(file,'>'))+1,*

*(instr(file,'</'))-(instr(file,'>'))-1)*

*as Title*

*from patent\_ipgb where file like '%invention-title%'*

**

***Invention-title field is extracted using Hive.***

***Text Parsing to get abstract:***

*select substr*

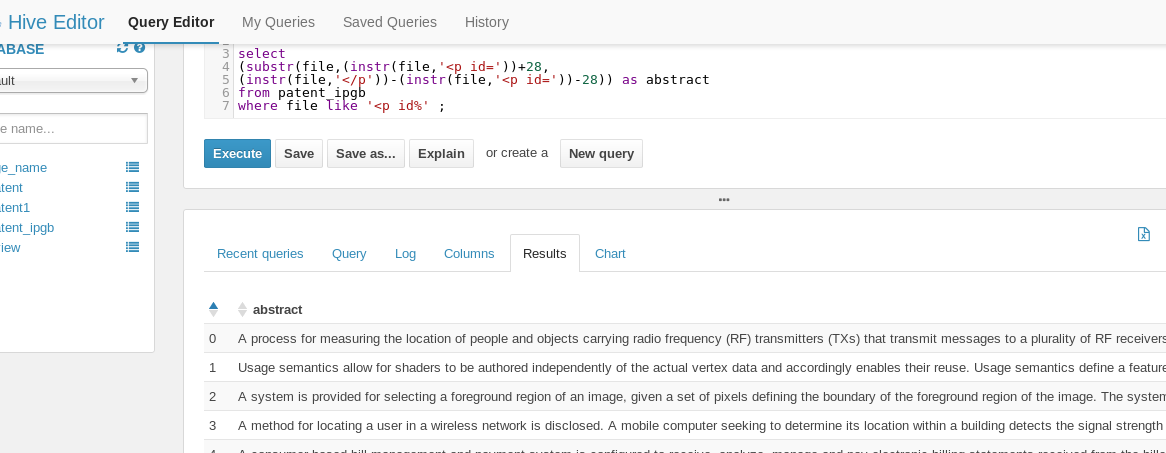
*(file,(instr(file,'<p id='))+28,*

*(instr(file,'</p'))-(instr(file,'<p id='))-28)*

*as abstract*

*from patent\_ipgb*

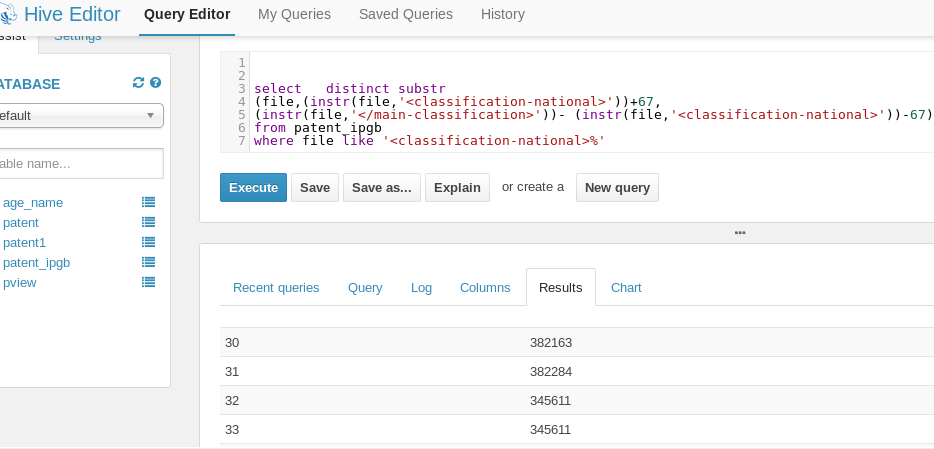
*where file like '<p id%' ;*

**

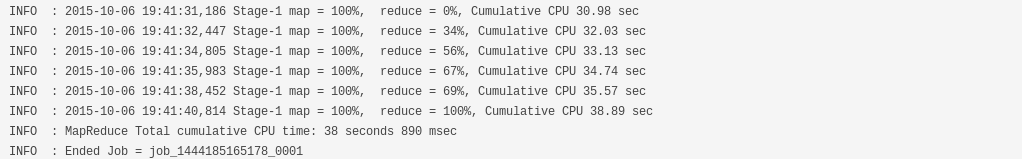
***Abstract field is extracted using Hive.***

***Text Parsing to get Class:***

*select  distinct substr  
(file,(instr(file,'<classification-national>'))+67,  
(instr(file,'</main-classification>'))- (instr(file,'<classification-national>'))-67)  
from patent\_ipgb   
where file like '<classification-national>%';*

**

***Log showing Map/Reducer Function in Hive:***



**Step3: Data Visualization using R and Tableau (R code to extract Top 50 words from Abstract column from text file)**

patentTxt <- VCorpus(DirSource("./textData"))

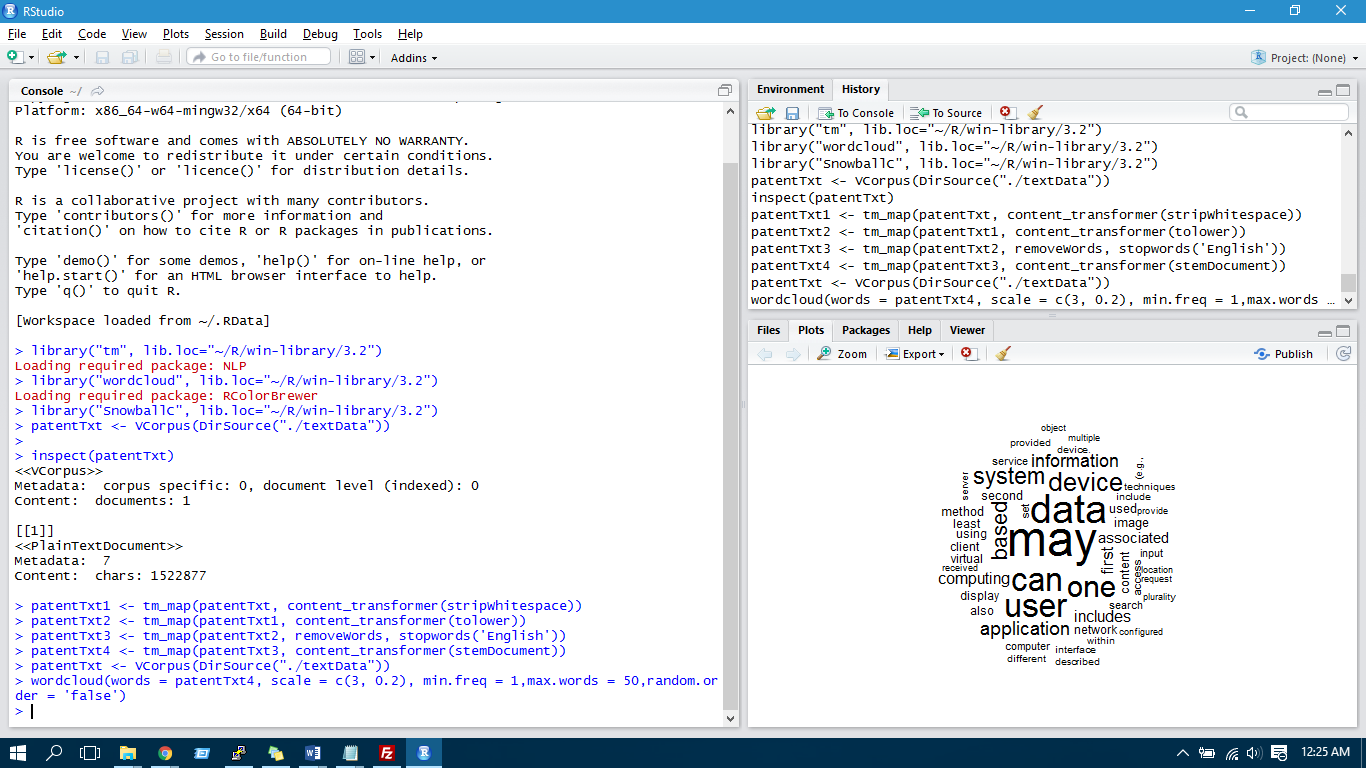
patentTxt1 <- tm\_map(patentTxt, content\_transformer(stripWhitespace))

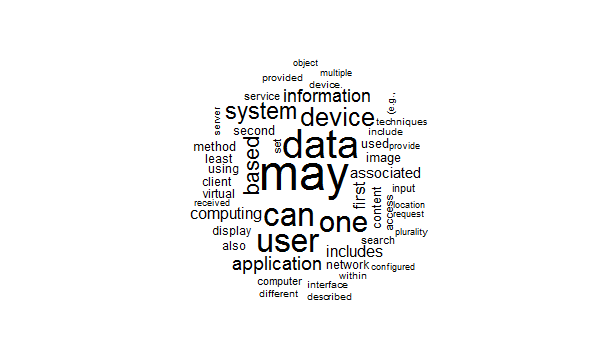
patentTxt2 <- tm\_map(patentTxt1, content\_transformer(tolower))

patentTxt3 <- tm\_map(patentTxt2, removeWords, stopwords('English'))

patentTxt4 <- tm\_map(patentTxt3, content\_transformer(stemDocument))

wordcloud(words = patentTxt4, scale = c(3, 0.2), min.freq = 1,max.words = 50,random.order = 'false')





Exported all the Data in PDF Format.

We imported the file in tableau and ran the visualizations.