# [https://avatars2.githubusercontent.com/u/4156894?v=3&s=100](http://www.calstatela.edu/centers/hipic) CIS 5200 Term Project Tutorial

#### **Authors:** [Jaydeep J Chopde](https://www.linkedin.com/in/jaydeep-chopde/) ; [Maitri S Shah](https://www.linkedin.com/in/maitri-shah-525013101/);

#### [Monika Mishra](https://www.linkedin.com/in/monika-mishra-8b2a4115/); [Pankti N Parikh](https://www.linkedin.com/in/panktiparikh/)

Rakshith Chandan Babu

#### **Instructor:**[Jongwook Woo](https://www.linkedin.com/in/jongwook-woo-7081a85)

#### **Date: 12/15/2018**

**Amazon Product Review Data Analysis**

**Term Project Group 2**

Jaydeep J Chopde, Maitri S Shah, Monika Mishra, Pankti N Parikh, Rakshith Chandan Babu

**Objectives**

In this lab you will analyze and visualize Amazon Product review data. Thus,

* You should learn how to download Amazon Product review data to the local systems in Oracle Cloud.
* Then, you will learn how to upload it to HDFS.
* You will figure out how to manipulate and analyze the data in HDFS using HiveQL.
* You will also practice how to visualize the result in Excel, Tableau and Powerbi.

# Introduction

Customer on any online shopping sites make purchasing decisions based on reviews and ratings. And so, it’s very important business to know various shopping patterns with a review dataset. Here you will do data analysis based on Amazon product review.

# Platform Specification

Cluster Version – Oracle Big Data Compute Edition

Number of Nodes – 5

Memory size – 150 GB

CPU – 20 vCPU

CPU speed – 2.20 GHz

HDFS capacity – 147 GB

Storage – 678 GB

# Prerequisites

Everything you need to go through the scripts and queries is already provisioned with the cluster. To export the analyzed data to Microsoft Excel, you must meet the following requirements:

* You must have an ip address to connect to Oracle Cloud .
* You must have **Microsoft Excel 2010**, **2013 or 2016** installed.
* You must have Tableau installed on your machine for visualizations.
* You must have an account with the PowerBI.

# 1. Connect to Oracle Cloud: Big Data Compute

You need to remotely access your Oracle Big Data that you executed in your Oracle Cloud account using *ssh*. For example, for the user name and ip address: **mmishra2**, you need to run the following with the ip address given:

|  |
| --- |
| $ ssh [mmishra2@ipadddress](mailto:jwoo5@129.150.128.177) |

When asked for password, type in your user name again and enter.

A screenshot of a cell phone

Description automatically generated

You are now connected to Oracle cloud.

# 2. Amazon review data loaded into Oracle Big Data

Below is the location of the Amazon product review data that is used for this sample. You can download the data file from amazon aws:

$ wget -O review\_de.tsv.gz https://s3.amazonaws.com/amazon-reviews-pds/tsv/amazon\_reviews\_multilingual\_DE\_v1\_00.tsv.gz

You should get something like this:

A screenshot of a cell phone

Description automatically generated

Similarly download other files as well.

$ wget -O review\_us.tsv.gz https://s3.amazonaws.com/amazon-reviews-pds/tsv/amazon\_reviews\_multilingual\_US\_v1\_00.tsv.gz

$ wget -O review\_uk.tsv.gz https://s3.amazonaws.com/amazon-reviews-pds/tsv/amazon\_reviews\_multilingual\_UK\_v1\_00.tsv.gz

$ wget -O review\_fr.tsv.gz https://s3.amazonaws.com/amazon-reviews-pds/tsv/amazon\_reviews\_multilingual\_FR\_v1\_00.tsv.gz

Then, you need to unzip the files.

$ gunzip review\_us.tsv.gz

$ gunzip review\_uk.tsv.gz

$ gunzip review\_de.tsv.gz

$ gunzip review\_fr.tsv.gz

You also need to download the US and Germany and French dictionary files using command as below:

$ wget -O dictionary\_us.tsv https://s3.amazonaws.com/hipicdatasets/dictionary.tsv

$ wget -O dictionary\_germany.zip http://www.ulliwaltinger.de/sentiment/GermanPolarityClues-2012.zip

$ wget -O dictionary\_fr.csv http://advanse.lirmm.fr/FEEL.csv

Then, you need to unzip dictionary\_germany.zip file and you will see 6 files are uncompressed under “GermanPolarityClues-2012” directory:

$ unzip dictionary\_germany.zip

$ ls GermanPolarityClues-2012



You will use three of the above files in the folder GermanPolarityClues-2012 for this lab.

# 3. Create directories in HDFS

Run the following commands for creating directories:

$ hdfs dfs -mkdir project

$ hdfs dfs -mkdir project/tables

$ hdfs dfs -mkdir project/tables/us

$ hdfs dfs -mkdir project/tables/uk

$ hdfs dfs -mkdir project/tables/de

$ hdfs dfs -mkdir project/tables/fr

$ hdfs dfs -mkdir project/tables/dictionary\_us

$ hdfs dfs -mkdir project/tables/dictionary\_ge

$ hdfs dfs -mkdir project/tables/dictionary\_fr

You can view the directories created by the following command:

$ hdfs dfs -ls project/tables

A screenshot of a cell phone

Description automatically generated

# 4. Put files in HDFS directories

# Run the following commands to put dictionaries into respective folders:

$ hdfs dfs -put dictionary\_us.tsv project/tables/dictionary\_us/

$ hdfs dfs -put GermanPolarityClues-2012/GermanPolarityClues-Negative-21042012.tsv project/tables/dictionary\_ge/

$ hdfs dfs -put GermanPolarityClues-2012/GermanPolarityClues-Neutral-21042012.tsv project/tables/dictionary\_ge/

$ hdfs dfs -put GermanPolarityClues-2012/GermanPolarityClues-Positive-21042012.tsv project/tables/dictionary\_ge/

$ hdfs dfs -put FEEL.csv project/tables/dictionary\_fr/

# You can run the following commands to check the files are there:

$ hdfs dfs -ls project/tables/dictionary\_ge/

$ hdfs dfs -ls project/tables/dictionary\_us/

$ hdfs dfs -ls project/tables/dictionary\_fr/



Similarly run the following commands to place data review files in the corresponding folders:

$ hdfs dfs -put review\_uk.tsv project/tables/uk/

$ hdfs dfs -put review\_us.tsv project/tables/us/

$ hdfs dfs -put review\_fr.tsv project/tables/fr/

$ hdfs dfs -put review\_de.tsv project/tables/de/

Run the following command to provide permission to the files under project folder:

hdfs dfs -chmod -R o+w project/

# 5. Creating Hive tables to query data

## The following Hive statement creates an external table that allows Hive to query data stored in HDFS. External tables preserve the data in the original file format, while allowing Hive to perform queries against the data within the file.

## Open another terminal and login into your account using ssh as in Step 1.

## Open **beeline** CLI (Command Line Shell Interface) that is equivalent to **hive** CLI environment as follows. **Beeline** is for multiple users’ access to Hive Server 2 of a Hadoop cluster. Press enter without any password when it asks for password.

**NOTE**: the following connect url is an example and it should be given by the instructor:

-bash-4.1$ beeline

WARNING: Use "yarn jar" to launch YARN applications. Beeline version 1.2.1000.2.4.2.0-258 by Apache Hive

beeline> **!connect jdbc:hive2://cis5200-bdcsce-4.compute- 608214094.oraclecloud.internal:2181,cis5200-bdcsce-2.compute- 608214094.oraclecloud.internal:2181,cis5200-bdcsce-3.compute- 608214094.oraclecloud.internal:2181/;serviceDiscoveryMode=zooKeeper;zooKeeper Namespace=hiveserver2?tez.queue.name=interactive bdcsce\_admin**

Connecting to jdbc:hive2://cis5200-bdcsce-4.compute- 608214094.oraclecloud.internal:2181,cis5200-bdcsce-2.compute- 608214094.oraclecloud.internal:2181,cis5200-bdcsce-3.compute- 608214094.oraclecloud.internal:2181/;serviceDiscoveryMode=zooKeeper;zooKeeper Namespace=hiveserver2?tez.queue.name=interactive

**Enter password** for jdbc:hive2://cis5200-bdcsce-4.compute- 608214094.oraclecloud.internal:2181,cis5200-bdcsce-2.compute- 608214094.oraclecloud.internal:2181,cis5200-bdcsce-3.compute- 608214094.oraclecloud.internal:2181/;serviceDiscoveryMode=zooKeeper;zooKeeper Namespace=hiveserver2?tez.queue.name=interactive:

Connected to: Apache Hive (version 1.2.1000.2.4.2.0-258)

Driver: Hive JDBC (version 1.2.1000.2.4.2.0-258) Transaction isolation: TRANSACTION\_REPEATABLE\_READ

0: jdbc:hive2://cis5200-bdcsce-4.compute-6082>

## **NOTE**: If you see “CLOSED” in the above beeline shell prompt, it is not connected to Hive Server2.

Now you have to create your database with your username to separate your tables with other users. For example, the user **mmishra2** should run the following:

**NOTE**: you have to use your username.

0: jdbc:hive2://cis5200-bdcsce-4.compute-6082> CREATE DATABASE mmishra2; No rows affected (0.277 seconds)

0: jdbc:hive2://cis5200-bdcsce-4.compute-6082> show DATABASES;

+----------------+--+

| database\_name |

+----------------+--+

| default |

| mmishra2 |

| ngupta8 |

| whu4 |

+----------------+--+

4 rows selected (0.232 seconds)

0: jdbc:hive2://cis5200-bdcsce-4.compute-6082> use mmishra2; No rows affected (0.184 seconds)

## In the beeline shell CLI, you need to copy and paste the following HiveQL code to create external tables**.**

**NOTE: Don’t forget to replace mmishra2 to your account name in the following HQL code.**

DROP TABLE IF EXISTS dictionary\_ge;

CREATE EXTERNAL TABLE if not exists dictionary\_ge (

word string,

word1 string,

misc string,

polarity string,

stemmed string,

misc1 string)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\t'

STORED AS TEXTFILE

LOCATION '/user/**mmishra2**/project/tables/dictionary\_ge';

DROP TABLE IF EXISTS dictionary\_us;

CREATE EXTERNAL TABLE if not exists dictionary\_us (

type string,

length int,

word string,

pos string,

stemmed string,

polarity string )

ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\t'

STORED AS TEXTFILE

LOCATION '/user/**mmishra2**/project/tables/dictionary\_us';

DROP TABLE IF EXISTS dictionary\_fr;

CREATE EXTERNAL TABLE if not exists dictionary\_fr (

word string, polarity string,

joy int, fear int, sadness int, anger int, surprise int,

disgust int)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\,'

STORED AS TEXTFILE

LOCATION '/user/**mmishra2**/project/tables/dictionary\_fr';

The above commands created dictionary\_us, dictionary\_fr and dictionary\_ge tables which will later be used for sentiment analysis.

You can query the result by running select statement:

select \* from dictionary\_ge limit 10;

select \* from dictionary\_us limit 10;

select \* from dictionary\_fr limit 10;

Now the product review data table will be created.

DROP TABLE IF EXISTS review\_fr;

CREATE EXTERNAL TABLE if not exists review\_fr (

marketplace string,

customer\_id int,

review\_id string,

product\_id string,

product\_parent int,

product\_title string,

product\_category string,

star\_rating int,

helpful\_votes int,

total\_votes int,

vine string,

verified\_purchase string,

review\_headline string,

review\_body string,

review\_date timestamp)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\t'

STORED AS TEXTFILE

LOCATION '/user/**mmishra2**/project/tables/fr';

DROP TABLE IF EXISTS review\_us;

CREATE EXTERNAL TABLE if not exists review\_us (

marketplace string,

customer\_id int,

review\_id string,

product\_id string,

product\_parent int,

product\_title string,

product\_category string,

star\_rating int,

helpful\_votes int,

total\_votes int,

vine string,

verified\_purchase string,

review\_headline string,

review\_body string,

review\_date timestamp)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\t'

STORED AS TEXTFILE

LOCATION '/user/**mmishra2**/project/tables/us';

DROP TABLE IF EXISTS review\_uk;

CREATE EXTERNAL TABLE if not exists review\_uk (

marketplace string,

customer\_id int,

review\_id string,

product\_id string,

product\_parent int,

product\_title string,

product\_category string,

star\_rating int,

helpful\_votes int,

total\_votes int,

vine string,

verified\_purchase string,

review\_headline string,

review\_body string,

review\_date timestamp)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\t'

STORED AS TEXTFILE

LOCATION '/user/**mmishra2**/project/tables/uk';

DROP TABLE IF EXISTS review\_de;

CREATE EXTERNAL TABLE if not exists review\_de (

marketplace string,

customer\_id int,

review\_id string,

product\_id string,

product\_parent int,

product\_title string,

product\_category string,

star\_rating int,

helpful\_votes int,

total\_votes int,

vine string,

verified\_purchase string,

review\_headline string,

review\_body string,

review\_date timestamp)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY '\t'

STORED AS TEXTFILE

LOCATION '/user/**mmishra2**/project/tables/de';

Then, in the beeline shell, you need to check if the tables are shown:

show tables;

0: jdbc:hive2://cis5200s3-bdcsce-4.compute-60> show tables;

+----------------+--+

| tab\_name |

+----------------+--+

| dictionary\_ge |

| dictionary\_us |

| review\_de |

| review\_fr |

| review\_uk |

| review\_us |

+----------------+--+

14 rows selected (0.208 seconds)

Now you can query the contents of the table:

select \* from review\_de limit 10;

## You will observe that the column review\_date is NULL for all the rows. This is because Hive supports reading alternative timestamp formats. To make it work, run the following Hive query:

alter table review\_de SET SERDEPROPERTIES ("timestamp.formats"="yyyy-MM-dd");

alter table review\_us SET SERDEPROPERTIES ("timestamp.formats"="yyyy-MM-dd");

alter table review\_uk SET SERDEPROPERTIES ("timestamp.formats"="yyyy-MM-dd");

alter table review\_fr SET SERDEPROPERTIES ("timestamp.formats"="yyyy-MM-dd");

## Now the review\_date column will show the correct date.

## You can see the structure of the table as well

describe review\_de;

A screenshot of text

Description automatically generated

# 6. Creating Hive Queries to Analyze Data

You will create a base table named review which will have all the data from the tables review\_us, review\_uk, review\_de and review\_fr. All the subsequent queries will be based on the new created table.

DROP TABLE IF EXISTS review;

CREATE TABLE review AS

select \* from review\_de where review\_id is not null and star\_rating is not null

union

select \* from review\_fr where review\_id is not null and star\_rating is not null

union

select \* from review\_uk where review\_id is not null and star\_rating is not null

union

select \* from review\_us where review\_id is not null and star\_rating is not null;

The ‘Where’ clause is used in the above table for cleaning any junk data.

You can query the table using following select statement:

select \* from review limit 5;

**1.** **The below query will return the review count group by star rating.**

select count(review\_id) count, star\_rating from review

group by star\_rating order by star\_rating;

A screenshot of a cell phone

Description automatically generated

Now a table will be created using this query and stored in HDFS for visualization at a later stage.

CREATE TABLE IF NOT EXISTS rating ROW FORMAT DELIMITED

FIELDS TERMINATED BY "," STORED AS TEXTFILE

LOCATION "/user/**mmishra2**/project/tables/one" AS

SELECT COUNT(review\_id)count ,star\_rating FROM review

GROUP BY star\_rating ORDER BY star\_rating;

Switch on to the first terminal. You can see the directory “one” has been created under project/tables and if you view the directory you can see a file has been placed there.

-bash-4.1$ hdfs dfs -ls project/tables/one/

Found 1 items

-rwxr-xrwx 2 bdcsce\_admin hdfs 47 2018-11-18 08:12 project/tables/one/000000\_0

You can view the contents of the file with the below command:

-bash-4.1$ hdfs dfs -cat project/tables/one/000000\_0

554283,1

383986,2

738964,3

1714006,4

6180622,5

**2. The below command will create table based on top ten highest number of reviews given by unique users over ten years.**

CREATE TABLE IF NOT EXISTS users ROW FORMAT DELIMITED

FIELDS TERMINATED BY "," STORED AS TEXTFILE

LOCATION "/user/**mmishra2**/project/tables/two" AS

SELECT COUNT(review\_id) count, customer\_id FROM review

GROUP BY customer\_id ORDER BY count DESC LIMIT 10;

This created a table “users” and stored it as a text file under project/tables/two directory.

**3. The below command will create table based on number of review count grouped by year and month**

CREATE TABLE IF NOT EXISTS review\_date ROW FORMAT DELIMITED

FIELDS TERMINATED BY "," STORED AS TEXTFILE

LOCATION "/user/**mmishra2**/project/tables/three" AS

SELECT COUNT(review\_id) count, YEAR(review\_date)year, MONTH(review\_date) month FROM review

GROUP BY YEAR(review\_date),MONTH(review\_date)

ORDER BY year, month;

This created a table “review\_date” and stored it as a text file under project/tables/three directory.

**4. The below command will create table based on review count by product category**

CREATE TABLE IF NOT EXISTS product ROW FORMAT DELIMITED

FIELDS TERMINATED BY "," STORED AS TEXTFILE

LOCATION "/user/**mmishra2**/project/tables/four" AS

SELECT COUNT(review\_id) count, product\_category FROM review

GROUP BY product\_category;

This created a table “product” and stored it as a text file under project/tables/four directory.

**5. The below command will create table for top ten popular products based on average rating of 5 and having maximum review count.**

CREATE TABLE IF NOT EXISTS popular ROW FORMAT DELIMITED

FIELDS TERMINATED BY "," STORED AS TEXTFILE

LOCATION "/user/**mmishra2**/project/tables/five" AS

SELECT COUNT(review\_id)count, AVG(star\_rating) rating,

product\_title, product\_category, marketplace FROM review

GROUP BY product\_title, product\_category, marketplace

HAVING AVG(star\_rating) = 5 ORDER BY count DESC LIMIT 10;

This created a table “popular” and stored it as a text file under project/tables/five directory.

**6. The below command will create table showing rating of product category “Baby” by country**

CREATE TABLE IF NOT EXISTS baby ROW FORMAT DELIMITED

FIELDS TERMINATED BY "," STORED AS TEXTFILE

LOCATION "/user/**mmishra2**/project/tables/seven" AS

SELECT review\_id, star\_rating, marketplace, review\_date

FROM review WHERE product\_category = 'Baby';

This created a table “baby” and stored it as a text file under project/tables/seven directory.

**7. Views and tables mentioned below will analyze overall sentiments for countries – US, UK, France and Germany.**

CREATE VIEW IF NOT EXISTS v1 AS

SELECT marketplace, review\_id, review\_body FROM review;

CREATE VIEW IF NOT EXISTS v2 AS

SELECT marketplace, review\_id, words FROM v1

LATERAL VIEW EXPLODE(SENTENCES(LOWER(review\_body))) dummy as words;

CREATE VIEW IF NOT EXISTS v3 AS

SELECT marketplace, review\_id, word FROM v2

LATERAL VIEW EXPLODE(words) dummy as word;

You will use US dictionary for countries US and UK. For Germany, you will use German dictionary and for France, you will use French dictionary.

CREATE VIEW IF NOT EXISTS v4 AS

SELECT marketplace, review\_id, v3.word ,

CASE d\_us.polarity

WHEN 'negative' THEN -1

WHEN 'positive' THEN 1

ELSE 0 END AS polarity

FROM v3 LEFT OUTER JOIN dictionary\_us d\_us on v3.word = d\_us.word

WHERE marketplace = 'US'

UNION

CREATE VIEW IF NOT EXISTS v4 AS

SELECT marketplace, review\_id, v3.word,

CASE d\_us.polarity

WHEN 'negative' THEN -1

WHEN 'positive' THEN 1

ELSE 0 END AS polarity

FROM v3 LEFT OUTER JOIN dictionary\_us d\_us on v3.word = d\_us.word

WHERE marketplace = 'UK'

UNION

CREATE VIEW IF NOT EXISTS v4 AS

SELECT marketplace, review\_id, v3.word,

CASE d\_us.polarity

WHEN 'negative' THEN -1

WHEN 'positive' THEN 1

ELSE 0 END AS polarity

FROM v3 LEFT OUTER JOIN dictionary\_fr d\_fr on v3.word = d\_us.word

WHERE marketplace = 'FR'

UNION

SELECT marketplace, review\_id, v3.word,

CASE d\_ge.polarity

WHEN 'negative' THEN -1

WHEN 'positive' THEN 1

ELSE 0 END AS polarity

FROM v3 LEFT OUTER JOIN dictionary\_ge d\_ge on v3.word = d\_ge.word

WHERE marketplace = 'DE';

CREATE view IF NOT EXISTS v5 AS

SELECT marketplace, review\_id,

CASE

WHEN SUM(polarity) > 0 THEN 'positive'

WHEN SUM(polarity) < 0 THEN 'negative'

ELSE 'neutral' END AS sentiment

FROM v4 GROUP BY marketplace, review\_id;

CREATE TABLE IF NOT EXISTS sentiment ROW FORMAT DELIMITED

FIELDS TERMINATED BY "," STORED AS TEXTFILE

LOCATION "/user/**mmishra2**/project/tables/All\_Countries" AS

SELECT \* FROM v5;

This created a sentiment based table and stored it as a text file under project/tables/All\_Countries directory.

**8. Ngram sentimental analysis for the least rated product in US**

The below query gives the five least rated product in US with review count greater than 50.

SELECT product\_id, product\_title, marketplace, product\_category, FORMAT\_NUMBER(AVG(star\_rating),2) AS avg\_rating, COUNT(\*) AS num

FROM review\_us

GROUP BY product\_id, product\_title, marketplace, product\_category

HAVING num > 50 ORDER BY avg\_rating limit 5;

Filtering out the least rated product based on the above query

SELECT product\_id, product\_title, marketplace, product\_category, FORMAT\_NUMBER(AVG(star\_rating),2) AS avg\_rating, COUNT(\*) AS num

FROM review\_us

WHERE product\_id = 'B00B5P37IG'

GROUP BY product\_id, product\_title, marketplace, product\_category;



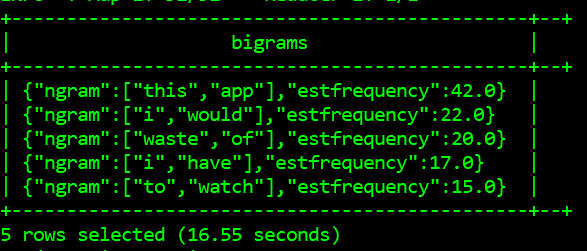
Using bigram for text analysis of the least rated product

SELECT EXPLODE(NGRAMS(SENTENCES(LOWER(review\_body)), 2, 5))

AS bigrams

FROM review\_us

WHERE product\_id= 'B00B5P37IG';



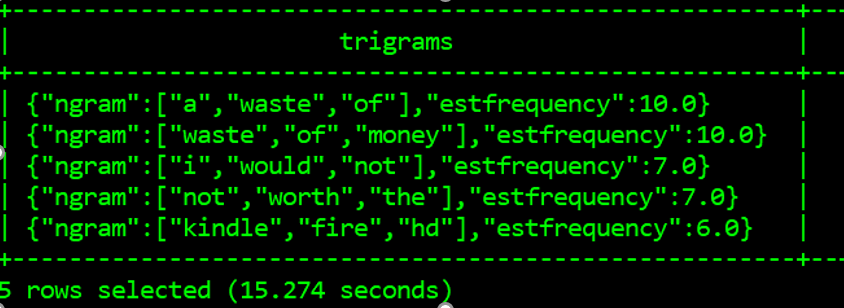
Using trigram for text analysis

SELECT EXPLODE(NGRAMS(SENTENCES(LOWER(review\_body)), 3, 5))

AS trigrams

FROM review\_us

WHERE product\_id= 'B00B5P37IG';



It can be seen that the commonly used word is “waste of”.

Retrieving sentences where the phrase “waste of” is used

SELECT review\_body

FROM review\_us

WHERE product\_id= 'B00B5P37IG'

AND review\_body LIKE '%waste of %'

LIMIT 3;

Based on the above query it can be concluded that the rating and review for the product “Live Tv Streaming (Kindle Tablet Edition)” is bad because of the following:

* Misleading description
* No free choice of streaming
* No support for local channels

On working over the above mentioned issues, the reviews and ratings for the product can be increased.

# 7. Downloading data into your PC

After the Hive tables are created, you can download it to your lab (or personal PC/Laptop) as follows.

1. Switch on to the first terminal connected to the Oracle cloud to download the output file

$ ssh mmishra2@ipaddress

mmishra2@ipaddress's password:

Run the following command to check if files are present:

|  |
| --- |
| -bash-4.1$ hdfs dfs -ls project/tables/one/  -bash-4.1$ hdfs dfs -ls project/tables/two/  -bash-4.1$ hdfs dfs -ls project/tables/three/  -bash-4.1$ hdfs dfs -ls project/tables/four/  -bash-4.1$ hdfs dfs -ls project/tables/five/  -bash-4.1$ hdfs dfs -ls project/tables/seven/  -bash-4.1$ hdfs dfs -ls project/tables/All\_Countries/ |

-bash-4.1$ hdfs dfs -ls project/tables/one/

Found 1 items

-rwxr-xrwx 2 bdcsce\_admin hdfs 47 2018-11-18 09:57 project/tables/one/000000\_0

You will see only one file named 000000\_0 is present in all the above folders except at project/tables/four/, project/tables/seven/ and project/tables/All\_Countries/. The latter locations have multiple and will need a merge.

1. Download the file to the local file systems and rename it

Since all the folders have the same name file 000000\_0, after downloading into local file system you will rename the file

-bash-4.1$ hdfs dfs -get project/tables/one/000000\_0

-bash-4.1$ ls

000000\_0 \_\_MACOSX

-bash-4.1$ mv 000000\_0 one

-bash-4.1$ ls

\_\_MACOSX one

Similarly, do it for others

-bash-4.1$ hdfs dfs -get project/tables/two/000000\_0

-bash-4.1$ mv 000000\_0 two

-bash-4.1$ hdfs dfs -get project/tables/three/000000\_0

-bash-4.1$ mv 000000\_0 three

-bash-4.1$ hdfs dfs -get project/tables/five/000000\_0

-bash-4.1$ mv 000000\_0 five

As the folder four, seven and All\_Countries have multiple files, you will need to merge it. The below command is used for merging multiple files into a single file.

-bash-4.1$ hdfs dfs -getmerge project/tables/four/ /home/mmishra2/four

-bash-4.1$ hdfs dfs -getmerge project/tables/seven/ /home/mmishra2/seven

-bash-4.1$ hdfs dfs -getmerge project/tables/All\_Countries/ /home/mmishra2/All\_Countries

1. Open another terminal in order to import the output file using your lab computer (or your PC/Laptop) - you have to download the file to your lab computer (or your PC/Laptop). For example, your output file at the oracle cloud server is located at /home/mmishra2/one and remotely copied to the file “one.csv”.

When asked for password, provide the password

$ scp mmishra2@ipaddress:/home/mmishra2/one one.csv

A screenshot of a cell phone

Description automatically generated

Similarly, do it for others

$ scp mmishra2@ipaddress:/home/mmishra2/two two.csv

$ scp mmishra2@ipaddress:/home/mmishra2/three three.csv

$ scp mmishra2@ipaddress:/home/mmishra2/four four.csv

$ scp mmishra2@ipaddress:/home/mmishra2/five five.csv

$ scp mmishra2@ipaddress:/home/mmishra2/seven seven.csv

$ scp mmishra2@ipaddress:/home/mmishra2/All\_Countries All\_Countries.csv

# 8. Visualizing data

**1. Review count group by star rating**

1. Open the one.csv file with Microsoft Excel and insert column heading as Review Count and Rating. Save the document.
2. Open a web browser and go to and sign in with your school account at: [https://app.powerbi.com](https://app.powerbi.com/)
3. Click on Local File and select the one.csv

A screenshot of a cell phone

Description automatically generated

1. Click on View dataset once the file is uploaded

A screenshot of a cell phone

Description automatically generated

1. Click on Donut chart under VISUALIZATIONS.

Drag Rating under Legend. Drag Review Count under Details and values

A screenshot of a cell phone

Description automatically generated

1. Click on the format under VISUALIZATIONS and switch on the Legend

You can customize colors and other features using format

A screenshot of a computer

Description automatically generated

**2. Number of reviews given by unique users**

1. Open the two.csv file with Microsoft Excel and insert column heading as Review Count and User. Save the document.
2. Open a web browser and go to and sign in with your school account at: [https://app.powerbi.com](https://app.powerbi.com/)
3. Click on Local File and select the two.csv
4. Click on View dataset once the file is uploaded
5. Click on Clustered column chart under VISUALIZATIONS.

Drag User under Axis and Color saturation. Drag Review Count under Value

A screenshot of a video game

Description automatically generated

1. Click on the format under VISUALIZATIONS and select type as “categorical” under X-Axis.

You can customize colors and other features using format

A screenshot of a cell phone

Description automatically generated

**3. Review count by year and month**

1. Open the three.csv file with Microsoft Excel and insert column heading as Count, Year and Month. Save the document.
2. Open a web browser and go to and sign in with your school account at: [https://app.powerbi.com](https://app.powerbi.com/)
3. Click on Local File and select the three.csv
4. Click on View dataset once the file is uploaded
5. Click on Clustered bar chart under VISUALIZATIONS.

Drag Year and month under Axis.

Drag month under legend.

Drag Count under Value

1. You will display values from 2010 to 2014 and so under filter section for Year select as per the below screenshot and click apply filter

A screenshot of a cell phone

Description automatically generated

The visualization will be like as below:

A screenshot of a computer screen

Description automatically generated

**4. Review count by product category**

1. Open the four.csv file with Microsoft Excel and insert column heading as Count and Product Category. Save the document in xlsx format.
2. Open the tableau in your machine
3. Click on Microsoft Excel under Connect and select the four.xlsx file

A screenshot of a cell phone

Description automatically generated

1. Click on the New worksheet at the bottom
2. Drag Product Category under Rows and Count under columns

Click on the packed bubbles at the extreme right

A picture containing screenshot

Description automatically generated

1. You can drag Product Category from Dimensions and Count from Measures on the extreme right to Marks which will customize your visualization

A screenshot of a cell phone

Description automatically generated

**5. Popular product based on average rating and review Count**

1. Open the five.csv file with Microsoft Excel and insert column heading as Count, Rating, Product, Category and Country. Save the document in xlsx format.
2. Open the tableau in your machine
3. Click on Microsoft Excel under Connect and select the five.xlsx file
4. Click on the New worksheet at the bottom
5. Drag Category, Product, Country and Rating under Rows and Count under columns

A screenshot of a cell phone

Description automatically generated

Click on the treemaps at the right

A screenshot of a social media post

Description automatically generated

1. You can drag count measures more than once to arrange it by colors and sizes. You can customize as per your wish

**6. Rating based geospatial representation of product category Baby**

1. Open the seven.csv file with Microsoft Excel and insert column heading as Review Id, Rating, Country and Time. Save in xlsx.
2. Open seven.xlsx in MS-excel. Go to Insert tab and click on 3D Map.



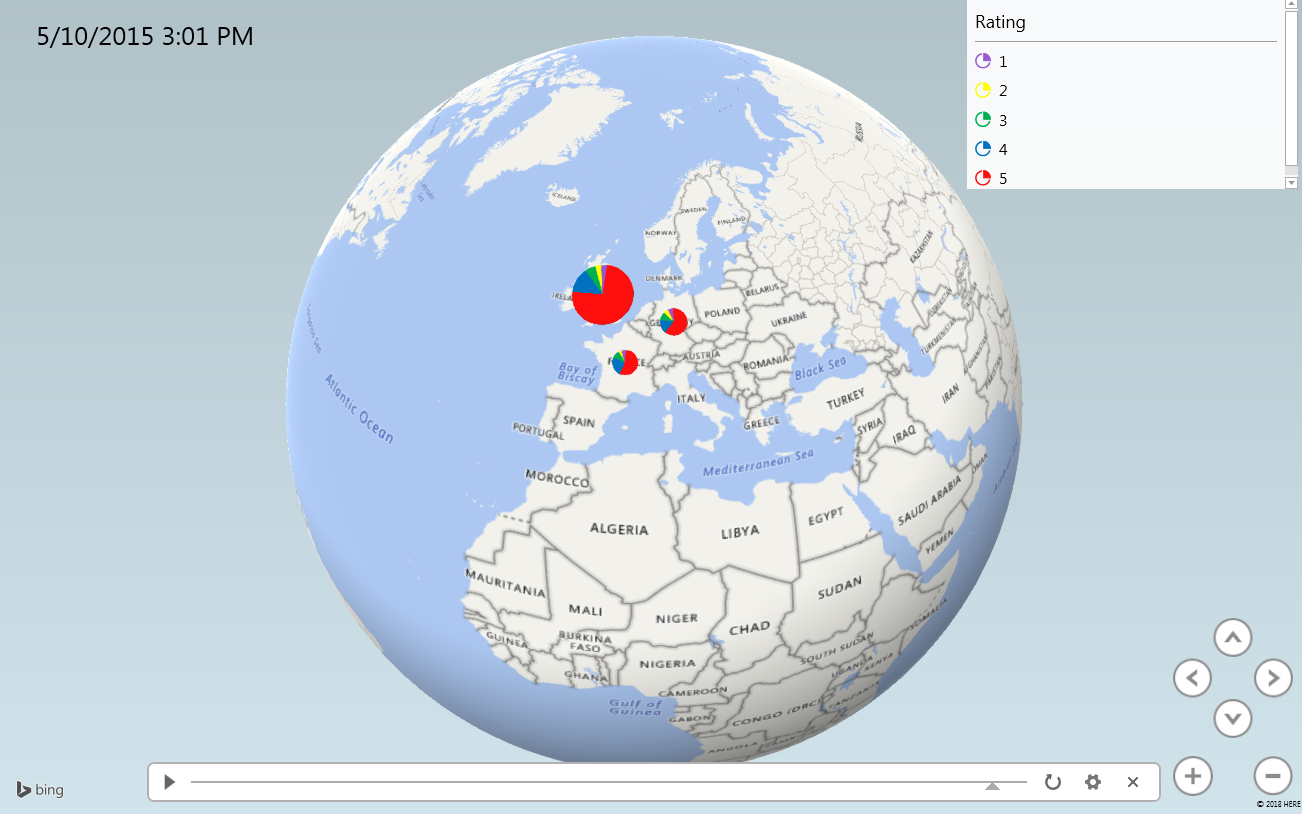
1. You will see the 3D map.

**NOTE:** If you don’t see the layer frame in the right side, you may select all data manually before opening 3D map

1. Add fields as shown in the picture below

A screenshot of a cell phone

Description automatically generated

1. You will get a view like below. You can click on **play** button to observe how much reviews have been generating



**7. Overall sentiment count by country**

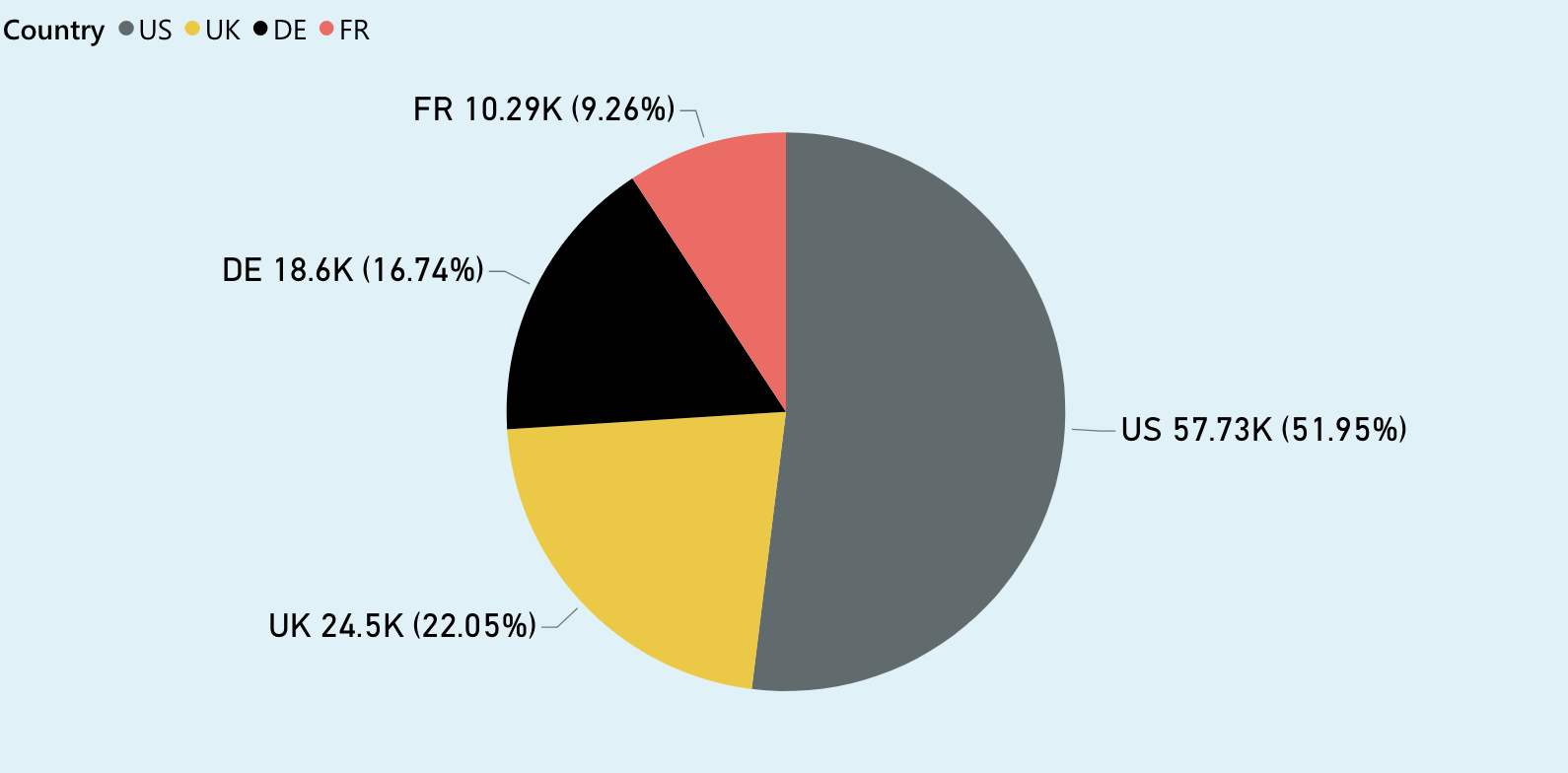
1. Open the All\_Countries.csv file with Microsoft Excel and insert column heading as Country, Review, Sentiment
2. Open a web browser and go to and sign in with your school account at: [https://app.powerbi.com](https://app.powerbi.com/)
3. Click on Local File and select the All\_Countries.csv
4. Click on View dataset once the file is uploaded
5. Click on Pie chart under VISUALIZATIONS.

Drag Country under Axis.

Drag sentiment under legend.

Drag Count of sentiment under Value

1. You can customize the colors as per your wish



**8. Overall sentiment analysis – positive, negative, neutral by country**

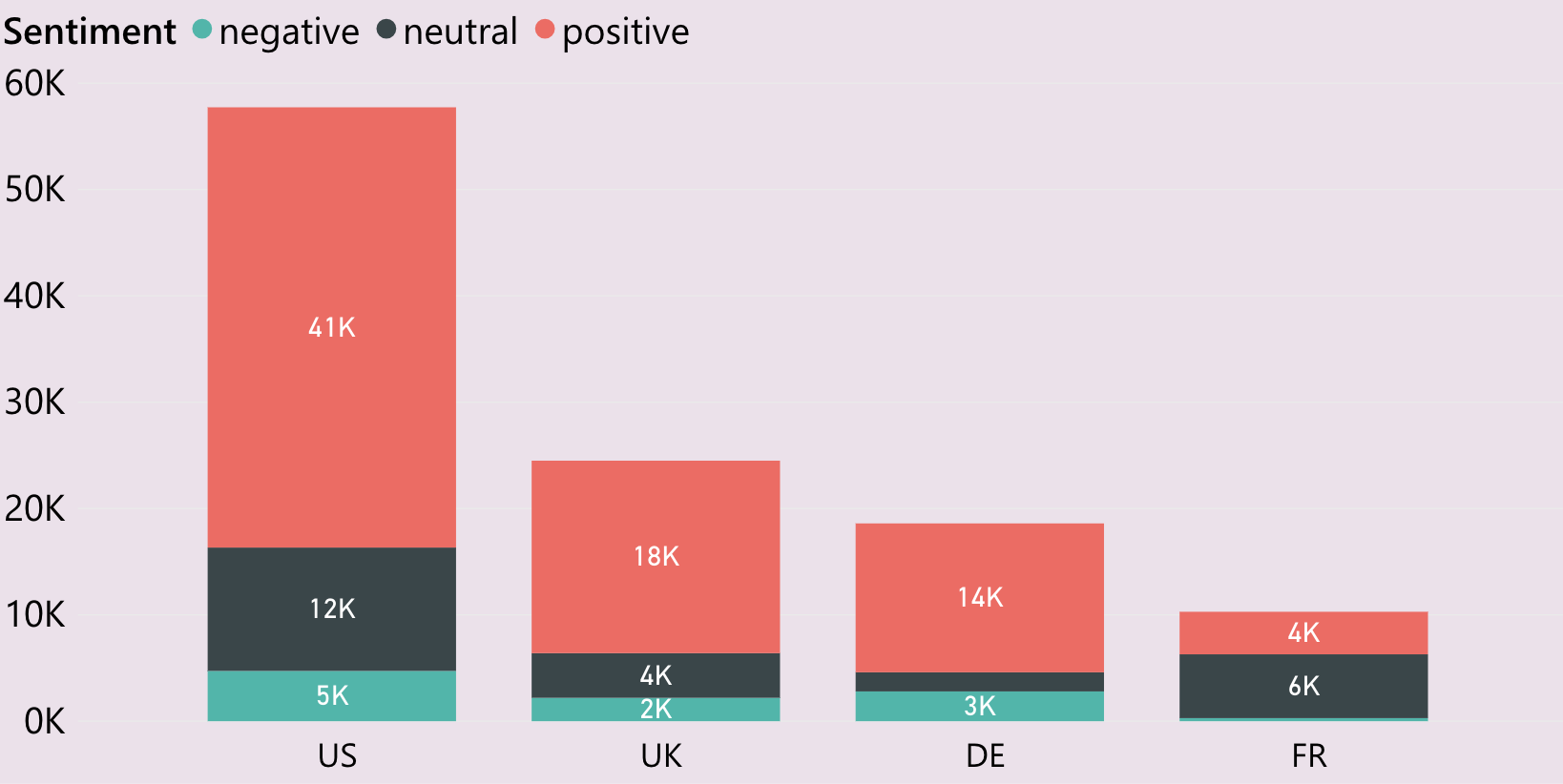
1. Use the same All\_Countries.csv file as above
2. Click on Stacked column chart under VISUALIZATIONS in Power BI.

Drag Country under Axis.

Drag sentiment under legend.

Drag Count of sentiment under Value

1. You can customize the colors as per your wish



# Summary

# In this tutorial you learned how Oracle Cloud Big Data can be used to analyze different patterns of raw data using Apache Hive. You went through a flow to understand how the raw data is first uploaded to HDFS, and then loaded to Hive tables for performing queries. And, you learned how to import the results of Hive queries into Microsoft Excel, tableau and powerbi. Finally, you learned how to create visualizations using tableau, powerbi and 3D Map chart in MS-excel.

# References

1. https://github.com/monika2403/mmishra2
2. https://s3.amazonaws.com/amazon-reviews-pds/tsv/amazon\_reviews\_multilingual\_US\_v1\_00.tsv.gz
3. https://s3.amazonaws.com/amazon-reviews-pds/tsv/amazon\_reviews\_multilingual\_DE\_v1\_00.tsv.gz
4. https://s3.amazonaws.com/amazon-reviews-pds/tsv/amazon\_reviews\_multilingual\_UK\_v1\_00.tsv.gz
5. https://s3.amazonaws.com/amazon-reviews-pds/tsv/amazon\_reviews\_multilingual\_FR\_v1\_00.tsv.gz
6. https://s3.amazonaws.com/hipicdatasets/dictionary.tsv
7. http://www.ulliwaltinger.de/sentiment/GermanPolarityClues-2012.zip
8. https://www.dropbox.com/home/cis4560\_5200Fall2018/labs
9. https://advance.lirmm.fr/FEEL.csv