CSCI 5408 – Data Management, Warehousing and Analytics

Assignment-2

Cloud Setup Steps:

I have followed the tutorials given in the labs for setting up the cloud environment.

Java Installation:

```
>sudo add-apt-repository -y ppa:webupd8team/java
```

```
>sudo apt-get update
```

Oracle Installation:

```
>sudo apt-get -y install openjdk-8-jdk-headless
```

Python Installation:

```
>sudo apt-get install python3
```

Apache Spark Installation:

```
>wget http://apache.forsale.plus/spark/spark-2.4.4/spark-2.4.4-bin-hadoop2.7.tgz
```

```
>sudo tar zxvf spark-2.4.4-bin-hadoop2.7.tgz
```

MongoDB Installation:

```
>sudo apt-get install gnupg [2].
```

```
>wget -qO - https://www.mongodb.org/static/pgp/server-4.2.asc | sudo apt-key add – [2].
```

```
>sudo apt-get install -y mongodb-org [2].
```

Importing files into MongoDB:

```
>sudo mongoimport --db mydb --collection tweets_data --type csv --file /home/ubuntu/TwitterData/tweets.csv –headerline
```

```
>sudo mongoimport --db mydb --collection articles --type csv --file /home/ubuntu/TwitterData/articles-cleaned.csv --headerline
```

Finding wordcount:

I have written a map reduce java script files and a python script to pass the map and reduce methods as parameters for map_reduce method [5]. By using regular expressions patterns, I have written separate scripts for mapping single words and double words and stored them in a file (tweets.txt for Twitter Data and articles.txt for NewsAPI Data). I have executed the script in the spark framework using the following command:

```
>sudo /home/ubuntu/server/spark-2.4.4-bin-hadoop2.7/bin/spark-submit wordcount.py
```

Data Extraction Process:

Twitter Data:

I have created a developer account in twitter and noted the consumer_key, consumer_secret, access_token, access_token_secret. I have used tweepy to extract the data [1]. By using the tokens and the search API and passing the search_words as parameters to get the tweets based on the given words and extracted more than 3000 tweets and stored them in a JSON file after performing cleaning of data (Please refer twitter.py file).

News API Data:

After creating the developer account and receiving the authorization key, I have sent the search words and pageSize of 100 (maximum) as parameters, I have retrieved the articles and performed certain cleaning operations using regular expressions [3]. I have stored the final output in a JSON file (please refer newsapidata.py).

Data Cleaning:

By using regular expressions, I have replaced all the special characters, URL’s with an empty string. To perform these, I have used python regular expressions package ‘re’.

Count of the given words:

After getting the count of all the single- and double-words using map_reduce, I have taken out the count of the given words by using simple string comparison of lines and stored them in separate file (Please refer finalwordcount.py).

References:

[1] Tweepy's Documentation on 'API', 'Authentication', 'Cursor', 'Extended Tweets', Available: https://tweepy.readthedocs.io/en/latest/index.html [Accessed on Oct. 25, 2019]

[2] Mongo DB's documentation , 'Install MongoDB Community Edition on Ubuntu', Available: https://docs.mongodb.com/manual/tutorial/install-mongodb-on-ubuntu/ [Accessed on Oct. 26, 2019]

[3] Parth Kinage's 'News API:Extracting News Headlines and Articles', Aug. 23, 2019, Available: https://python.gotrained.com/news-api/ [Accessed on Oct. 27, 2019]

[4] Martha Morrissey, Leah Wasser, Jeremey Diaz, Jenny Palomino's, 'Analyze Word Frequency Counts Using Twitter Data and Tweepy in Python', Updated Sept. 03, 2019, Available: https://python.gotrained.com/news-api/ [Accessed on Oct. 28, 2019]

[5] TutorialKart's , 'MongoDB Map Reduce', Available: https://www.tutorialkart.com/mongodb/mongodb-map-reduce/ [Accessed on Nov. 1, 2019]