**LAB2: DATA AGGREGATION, BIG DATA ANALYSIS AND VISUALIZATION**

**Submitted BY:**

**Ravi Malik (Person No: 50247302)**

**Abhinav Kumar (Person No: 50247660)**

**OVERVIEW**: In this lab, we will expand our skills in data exploration developed in Lab1 and enhance them by adding big data analytics and visualization skills. This document describes Lab2: Data Aggregation, Big Data Analysis and Visualization, involves (i) data aggregation from more than one source using the APIs (Application programming interface) exposed by data sources, (ii) Applying classical big data analytic method of MapReduce to the unstructured data collected, and (iii) building a visualization data product

Procedures and Tasks :

* **Part 1** of the lab is present inside the directory : **Lab 2 Part 1 Python/**

1. Lab 2 Part 1 Section 3.ipynb
2. Lab 2 Part 1 Section 4.ipynb
3. Lab 2 Part 1 Section 5.ipynb

* **Part 2:** We collected data about from at least two sources, one opinion-based social media in twitter, and research data in New York Times, for the same topic or key phrase. Process the two data sets collected individually using classical big data methods. Compare the outcomes using popular visualization methods.

**Topic : Immigration**

For this we used the following scripts :

1. Lab 2 MapReduce /fetchTweets.py
2. Lab 2 MapReduce /fetchNewsArticles.py
3. Lab 2 MapReduce /1\_twitter\_count.py

We got the following output files :

1. **Lab 2 MapReduce /TwitterData/tweets.json** : Contains the dictionary of the collected tweets
2. **Lab 2 MapReduce /TwitterData/tweets.txt** : Contains all the raw tweets and retweets on the topic “Immigration”
3. **Lab 2 MapReduce /NewsData/news\_data\_raw.txt :** Contains the dictionary(metadata) of the collected articles
4. **Lab 2 MapReduce /NewsData/news\_data.txt** : Contains all the news articles(500-600)

* The next task was to run MapReduce on the collected tweets and news Articles. We did this using Hadoop and Python .

The following python file contains the code for calculating the wordcount:

* + - **Lab 2 MapReduce \wc.py**

We got the following output files:

1. **Lab 2 MapReduce \output\_tweets.txt :** File for collected words from the tweets and their counts
2. **Lab 2 MapReduce \output\_news.txt** : File for collected words from the tweets and their counts

* The next task was to design a web page and feed the results by embedding d3.js code (with replaceable worldclouds) in it, finalizing the display of results.

The code for the generated wordcloud is present in the following directory:

**Lab 2 MapReduce\d3-wordcloud-master\example**

1. example.html : It contains the html code which displays the wordcloud having d3.js embedded in it
2. news.words.js : It is the d3.js file for visualizing words from news articles
3. tweets.words.js : It is the d3.js file for visualizing words from tweets

* Using the smallest data sets we collected we analyze each set (Twitter and News) word co-occurrence for top-ten words .Assuming context for co-occurrence is the “tweet” in the case of TwitterData, and the paragraph of the news article in the NewsData, the “map” function emits <word, co-occurring word> and your “reduce” function collates the co-occurrences and outputs them in a suitable format.

The python file which did this task was :

* + - **Lab 2 MapReduce \wc\_coocurence.py**

The following output files were generated :

1. **Lab 2 MapReduce \output\_coocurence\_news.txt:** File for collected words and their coocurences from the news articles and their counts
2. **Lab 2 MapReduce \output\_coocurence\_news.json** : Metadata for output\_coocurence\_news.txt
3. **Lab 2 MapReduce \output\_coocurence\_tweets.txt** : File for collected words and their coocurences from the tweets and their counts
4. **Lab 2 MapReduce \output\_coocurence\_tweets.json** : Metadata for output\_coocurence\_tweets.txt

We visualize the output by constructing a bar plot using d3.js .The length of the bars vary according to the count of the words.

The files are stored in the directory :

**Lab 2 MapReduce\d3-wordcloud-master\coocurences**

1. index.html : The html file which displays the barplot and contains d3.js embedded in it.
2. Book.csv : The input csv file which contains the top ten words.