CSE 587: Lab 2 Report

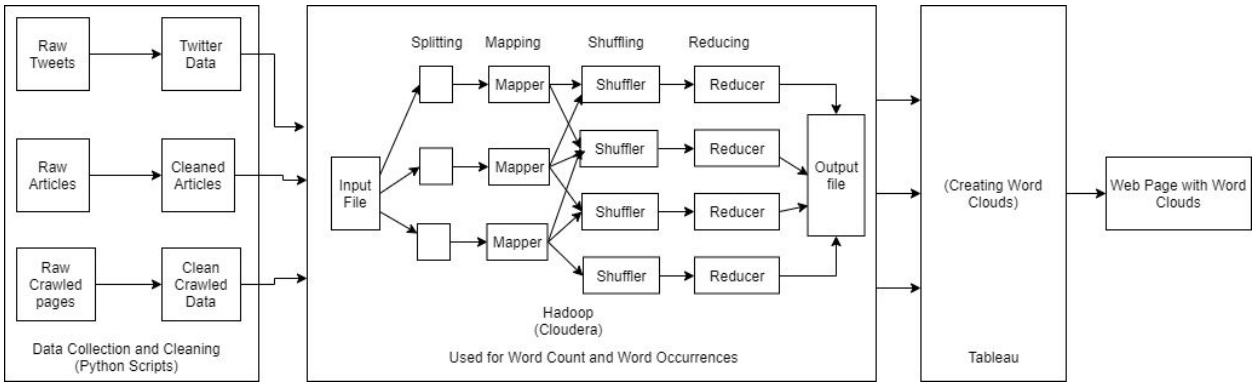
Team:

- Aarti Harindran (UBITName: aartitan)
- Dhanashree Solanke (UBITName: ddsolank)

1. Introduction

The goal of the project is to perform big data analytics on data fetched from sources like Twitter, NY Times, and Common Craw using API's and analyse results to understand the trends in the data by extracting top occuring words and their co-occurrences with others.

2. Architecture Diagram



3. Hadoop Setup

We are using Cloudera as a platform to perform all operations in Hadoop. The steps for which, are:

- Download VM Workstation
- Download Cloudera Quickstart VM image
- Mount and load the image on VM Workstation
- All the hadoop and python installations are by default present in the image

We can directly start running Hadoop operations

4. Data Collection and Cleaning

Our topic in Data Collection is 'Mental Health' in United States. The subtopics include attention deficiency hyperactivity disorder, post traumatic stress disorder, clinical depression, bipolar disorder and anxiety disorder.

- **Twitter Data**

  We have collected above 20000 tweets in total, with the count for each sub-topic being above 5000.

  We are using the Python Library called Twython which is a streaming API to extract tweets.

We have filtered the tweets by setting a search topic relevant to our keywords and filtering the data based on creation time and geographical location.
We have then stripped the tweets by removing emoticons, punctuations, URLs, numbers and stop words followed by stemming and lemmatization of the words.
We then write stripped tweets into a final file, with each tweet occurring on a new line.

- **NY Times Data**
  We have collected 500 articles in total (100 each for our five subtopics)
  We have used Python as a language for scripting to collect data
  The steps included are:
  - Register for NY Times Search API to get api-key
  - By including search keywords, we fetch all the urls of articles which are related to our topic
  - Then we perform scraping on those urls to store text data by using Beautifulsoup
  - We are also performing cleaning of data by removing numbers and punctuations from text and converting it into lower case. Then we remove stopwords and lemmatize words using nltk library.
  - Finally we dump one article as one line in our file.

- **Common Crawling**
  We have collected 800 articles in total for all subtopics.
  We are using March 2019 data for our Common Crawling.
  The steps are as follows:
  - By using Search option in Common Crawl March 2019, we first find out the urls which are related to our topics. In our case, we have used https://www.nimh.nih.gov/ which is the official site for national institute of mental health.
  - Above search returns us the related urls and the file offset for warc files
  - We then load the gz files and and save the urls
  - We scrape through all urls using Beautifulsoup and save the data by cleaning it
  - Cleaning procedure again involves removing numbers, punctuations, stop words and lemmatization and converting text to lower to finally dump one article per line in file.

5. **Mappers and Reducers**
   The task of the project includes calculating top used words and finding top word co-occurrences. We have written individual mappers and reducers to perform both.
   For running mapreduce in Cloudera image, we have used hadoop-streaming.jar. We have uploaded the fetched data into hdfs file structure and then performed the jobs.

Word Count:
The mapper will read tweet/article/page one by one and emit each word with count as 1
The reducer will read the words and the counts in sorted order and for each distinct word output the sum which is calculated by adding the count of same words.
Output File:

```
bargmann    1
barhaim 1
barke   5
barker  3
barlow  1
barnard 1
```

Word Co-occurrences:
The mapper is given the list of top 10 words (with highest count in decreasing order) from the above word count job.
The mapper then emits the pair of words(one of the top 10 words, other words in text) in a single tweet/article/page with count as 1
The reducer will consider pair of words as a single key and calculate the sum by adding the count for same keys and output the same.
The output files are then used for creating appropriate data visualizations.
Output File:

```
ability,clinical    82
ability,disorder    101
ability,health  123
ability,human   123
ability,mental  123
ability,program 63
ability,science 123
ability,study   96
```

Running Jobs
We divide our collected data into a subset and a superset to see the convergence between both. In twitter, the small data consists of one day data (around 5k tweets) on all subtopics whereas big data has 7 days data (20k tweets). In NY Times data, we have 100 articles as our small data and 500 in our big data. In Common crawl, we have 100 pages as small data and 800 pages as our big data.
We, pass these data files to our word count and word co-occurrences mapreduce jobs. Thus, we'll have 12 output files in total. We then go on plotting these files as a word cloud in Tableau.

6. Data Visualizations in Tableau

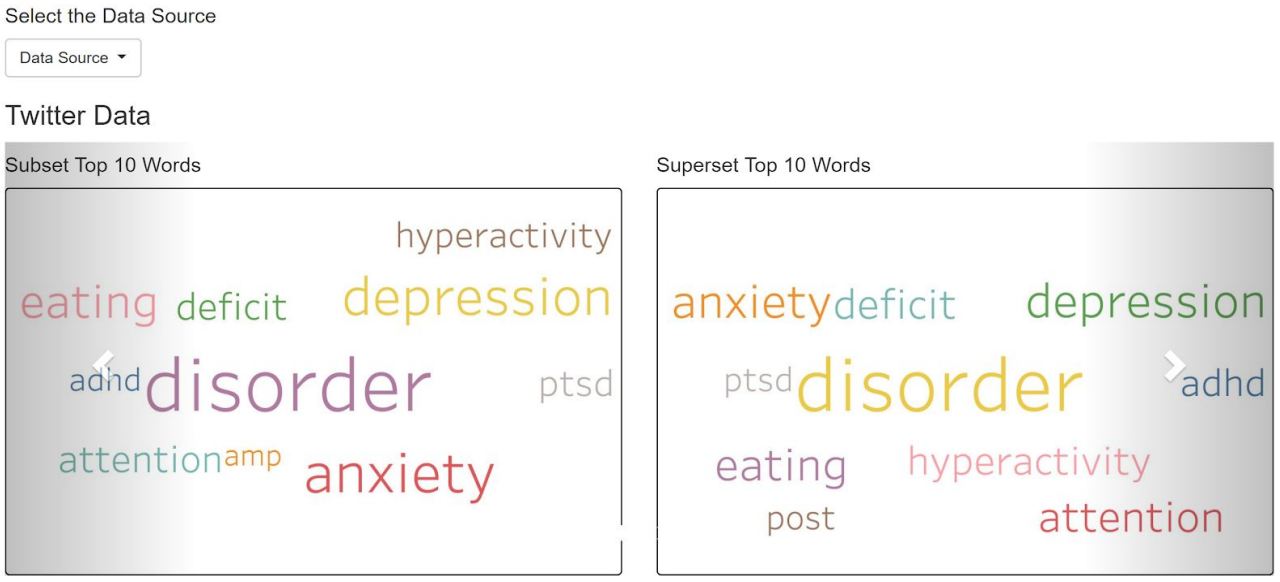
In Tableau, we create a book by selecting the data source as text file created by the reducers. We can create a word cloud by setting the actual word as the text label and the count as an attribute to be plotted. We can save the book or export the chart as an image. Thus, all word clouds are created using Tableau.

bigdataNYTimesWordCoOccur_N10 - NY Times top 10 co-occurring words of 500 articles
bigdataNYTimesWordCoOccur_N100 - NY Times top 100 co-occurring words of 500 articles
bigdataNYTimesWordCount_N10 -NY Times top 10 words of 500 articles
bigdataNYTimesWordCount_N100 - NY Times top 100 words of 500 articles
bigdataTwitterWordCoOccur_N10 - Twitter 7 day collections top 10 co-occurring words
bigdataTwitterWordCoOccur_N100 - Twitter 7 day collections top 10 co-occurring words
bigdataTwitterWordCount_N10 - Twitter 7 day collections top 10 words
bigdataTwitterWordCount_N100 - Twitter 7 day collections top 100 words
smalldataNYTimesWordCoOccur_N10 - Twitter 1 day collection top 10 co-occurring words
smalldataNYTimesWordCoOccur_N100 - Twitter 1 day collection top 100 co-occurring words
smalldataNYTimesWordCount_N10 - NY Times top 10 words for 100 articles
smalldataNYTimesWordCount_N100 - NY Times top 100 words for 100 articles
smalldataTwitterWordCoOccur_N10 - Twitter top 10 co-occurring words for 1 day collection
smalldataTwitterWordCoOccur_N100 - Twitter top 100 co-occurring words for 1 day collection
smalldataTwitterWordCount_N10 - Twitter top 10 words for 1 day collection
smalldataTwitterWordCount_N100 - Twitter top 100 words for 1 day collection
ccbigWordCount_N10 - Common Crawl top 10 words for 800 articles
ccbigWordCount_N100 - Common Crawl top 100 words for 800 articles
ccbigWordCoOccur_N10 - Common Crawl top 10 co-occurring words for 800 articles
ccbigWordCoOccur_N100 - Common Crawl top 100 co-occurring words for 800 articles
ccSmallWordCount_N10 - Common Crawl top 100 words for 100 articles
ccSmallWordCount_N100 - Common Crawl top 100 words for 100 articles
ccSmallWordCoOccur_N10 - Common Crawl top 10 co-occurring words for 100 articles
ccSmallWordCoOccur_N100 - Common Crawl top 100 co-occurring words for 100 articles

7. **Interactive Web Page**

   The comparison and analysis of different word clouds can be seen in the web page, which displays the word clouds by selected value of data source. Here, for a particular data source we compare the top 10 words, top 100 words, top 10 word co-occurrences and top 100 word co-occurrences and can see the convergence between them.

Twitter Comparison for the top 10 words occurring in one day of tweets vs seven days of tweets.



Twitter Comparison for the top 100 words occurring in one day of tweets vs seven days of tweets.