2019

Data intensive computing – Lab 2

REPORT

DISHA MEHRA - dishameh - 50288911

RAVI TEJA SUNKARA – rsunkara - 50292191

Introduction

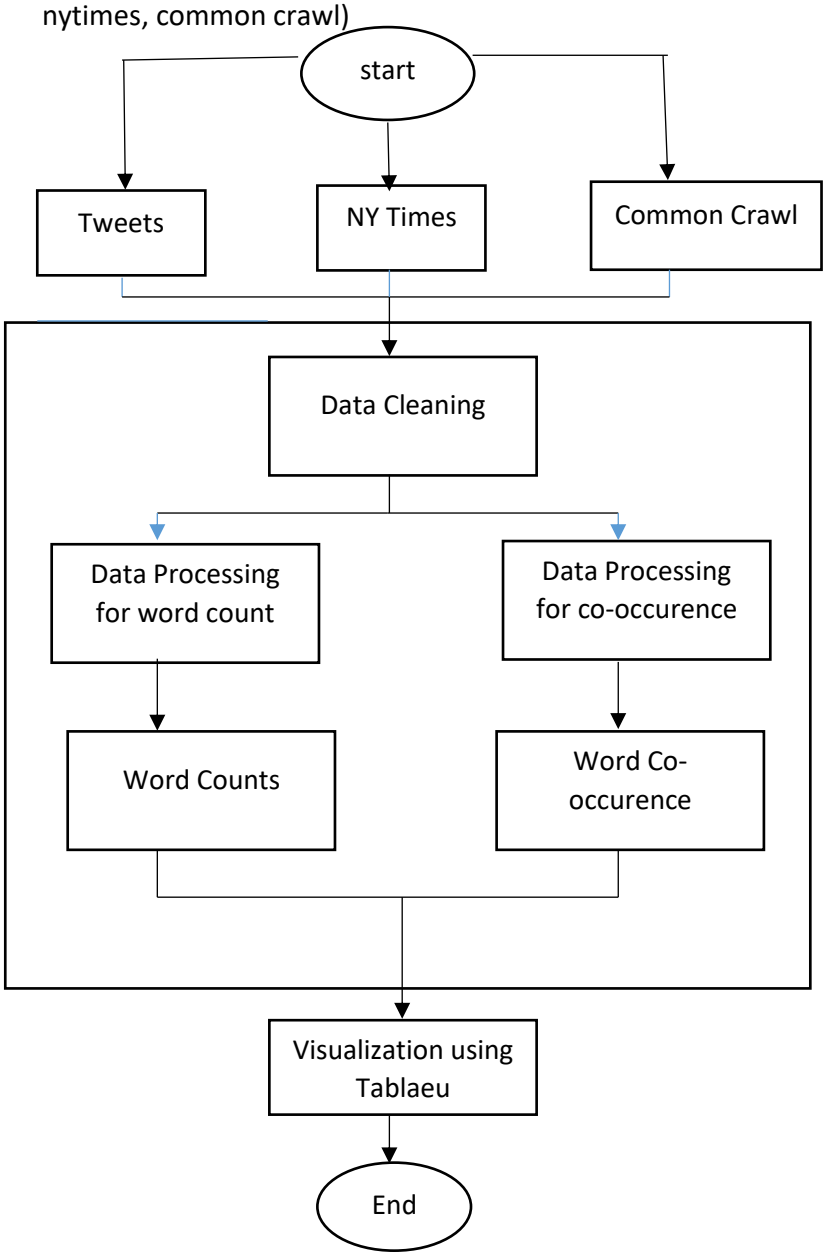
Business is a topic that is always in constant focus throughout the world and especially in a developed country like United States of America, as changes here will have significant ripple effects. The subtopics were chosen based on nytimes.com sub sections under business.

Topic – Business; **Subtopics** – Markets, Economy, Money, Entrepreneur, Energy

Implementation

We start by gathering the tweets from Twitter, articles from NY, articles from Common Crawl. Then we cleaned the data to gather the meaningful text data. Cleaned data is then processed using Map reduce in Hadoop to find the top 10 most occurring words and the co-occurring pairs. We used Cloudera Quickstart Docker image to implement single-node deployment of Hadoop. After the processing of big data using Hadoop, we visualized the results using Tableau to display the top 10 most occurring words and top 10 co-occurring words.

- Small data – used data of subtopic 'money'
- Big data – aggregated data which includes all the data from each source (twitter, nytimes, common crawl)



Data Collection

1. New York Times Articles
    - Details of all the articles were fetched using Article Search API provided by NY Times.
    - Using request and Beautiful Soup packages each article is scrapped using the article urls obtained from the article search API.
    - Articles were combined and saved based on subtopics. A nyt_final.txt file was created to combine the data of all the sub topics.
2. Tweets from Twitter
    - All tweets are fetched using rtweet package in R.
    - Tweets were collected based on subtopics. We used following search strings to collect tweets from United States. (business, markets, economy, money, entrepreneur, energy)
    - UTF-8 encoding was used while saving the tweets in a file.
    - Tweets for each subtopic are stored in separate text files. An aggregated text file has been created to represent the entire topic.
3. Common Crawl Articles
    - Common crawl index API was used to search for articles based on a website.
    - 'warcio' library in python 3 was used to retrieve the urls from each of WARC files obtained from indexer. The urls were filtered and then response and beautiful soup packages were used to obtain the article data. Articles which contained the keywords were saved.
    - Data was scraped from four different sites – forbes.com, cnbc.com, cnn.com, wsj.com and is stored in different text files using 'UTF-8' encoding.
    - A combined text file was created of all the data.

Word Count using MapReduce Framework

NLTK library in python 3 was used to remove stop words and create tokens. Coluder Quickstart Docker image by default comes with Cent OS 6 and python version 2.6.6. In order to implement the mapper and reducer in python 3, python 3.4 was installed and then pip3 was installed. The NLTK data was downloaded to /user/nltk_data folder in Hadoop.

We use the MapReduce framework to count the number of times a word occurred in the tweets or articles. We do this to find the most frequent words which capture the essence of the topic. Mapper and Reducer are implemented as follows:

- **Mapper**: In the Mapper, we cleaned the data to only keep the useful words. We replaced strings, punctuations, html tags, emojis using re.sub() function in python 3. Stopwords, downloaded from nltk library were used to remove the stop words. Tokens were created using the tokenize function.
  The Mapper then emits (outputs) a key value pair for each word in the article or tweet. The key in this case is the word and value is 1. I will later, in Reducer, aggregate these 1 for every key (unique word) to find the count of that word.
- **Reducer**: Output of the mapper is sent to the reducer. The reducer collects the <key, value> pairs which have same key and the all the values are summed up for a key. This generates the count for that key. The reducer then emits the total count of that key.

Co-occurrence using MapReduce Framework

The goal is to find the pair of words which occur most frequently together. Pairs of words are taken at a time and are passed through mapper and reducer.

- **Mapper**: In the first step, we clean the data by replacing various redundant strings as in word count. It is then followed by removing stop words and other common words which don't reflect the data. Then two words at a time are selected and <word-pair, 1> in <key, value> format are emitted by Mapper.
- **Reducer**: In the reducer, all the keys are collected of the pairs emitted by mapper. All the values (1) of a given key (word-pair) are aggregated to get the count of a given word-pair or co-occurring words. Reducer gives this as output. The procedure is applied to every unique word-pair sent as key. The resulting output gives the count/frequency of co-occurring words.

Results and Visualization using Tableau

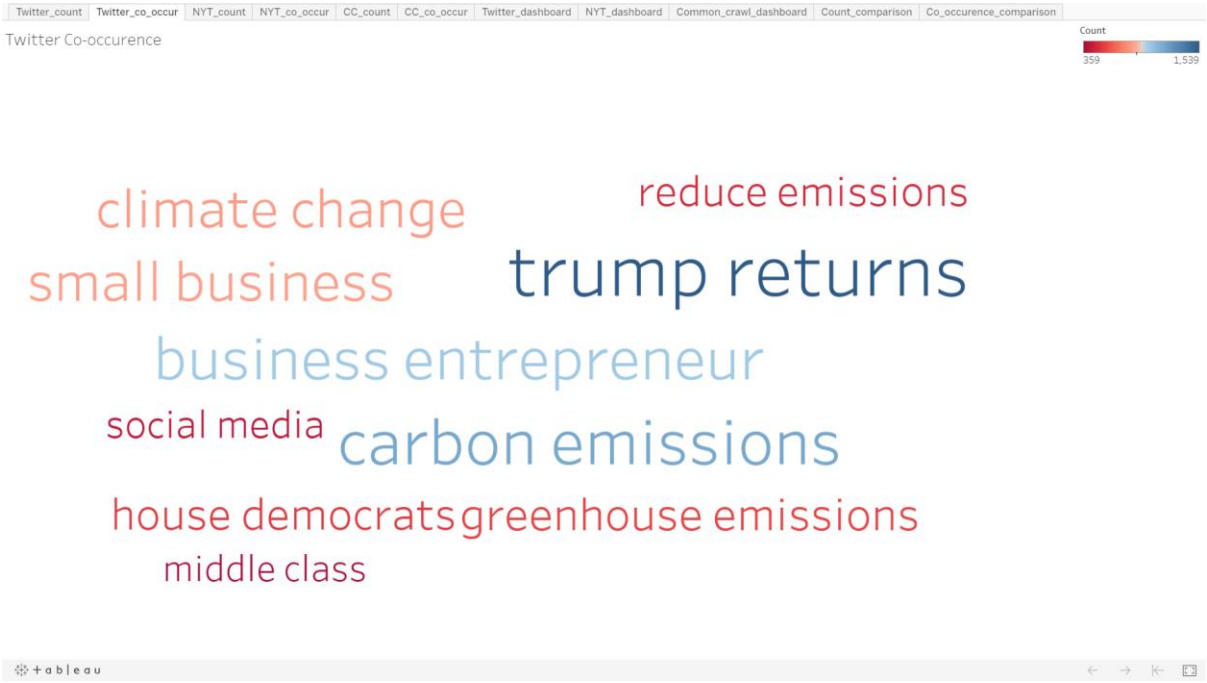
We displayed the top 10 most occurring words and co-occurring words using Tableau for Big data and Small data.

The commonly occurring words among NY Times articles, Twitter and Common Crawl data converged with business among the top 10 words. While the NY Times articles and tweets converged over many words like business, people, money, time and trump among the top 10 words.

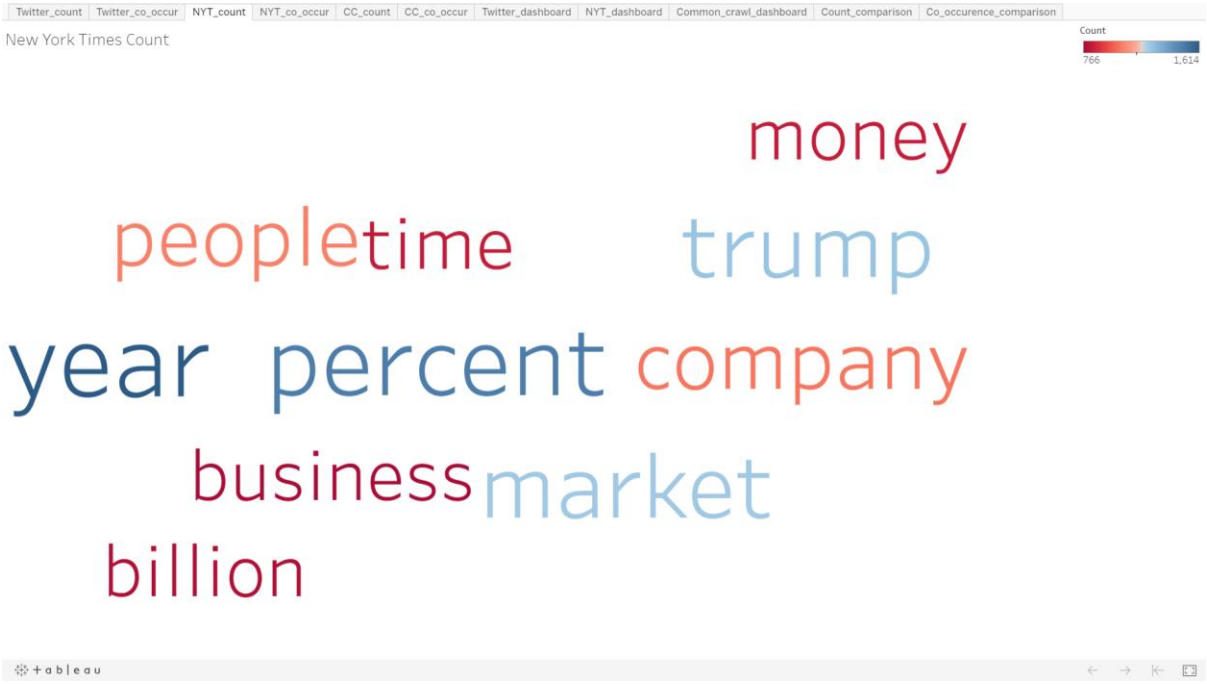
Big Data

Twitter Word Count Visualization

Twitter Co-Occurrence Visualization



New York Times Word Count Visualization

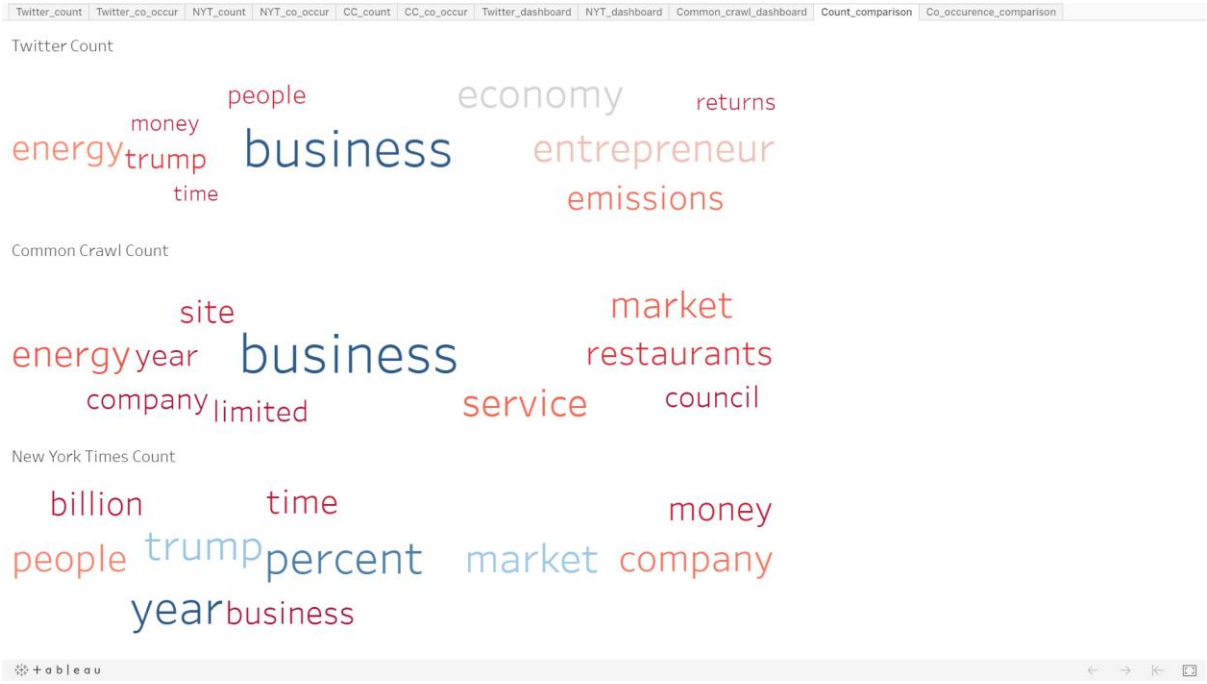


New York Times Co-Occurrence Visualization

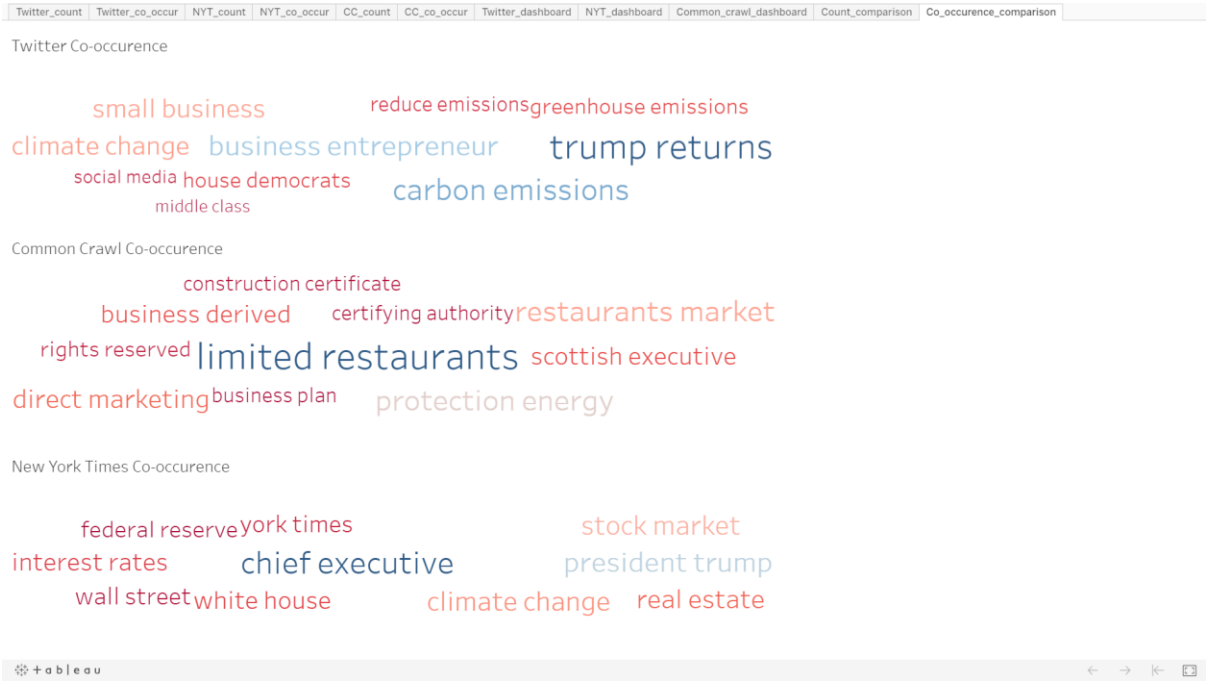
Common Crawl Word Count Visualization

Common Crawl Co-Occurrence Visualization

Word Cloud Comparison of word count for Twitter, NY Times and Common Crawl



Word Cloud Comparison of Co-Occurrence for Twitter, NY Times and Common Crawl

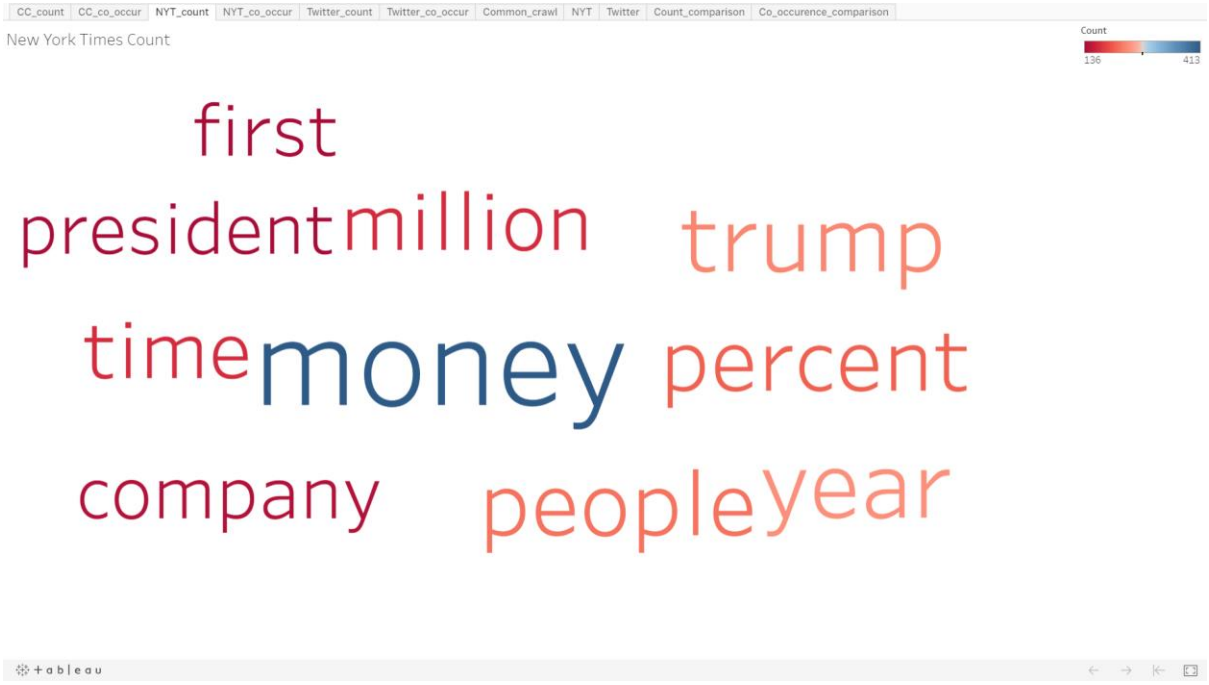


Small data

Twitter word cloud of top 10 words

Twitter word cloud of top 10 co-occurring words

New York Times word cloud of top 10 most occurring words



New York Times word cloud of top 10 co-occurring words

Common Crawl word cloud of top 10 frequent words

Common Crawl word cloud of top 10 most co-occurring words

Commands

Things to do before running our mapper and reducer

1. Install python 3.4 on the docker container
   https://tecadmin.net/install-python-3-4-on-centos-rhel-fedora/
2. Install pip3
   https://stackoverflow.com/questions/32618686/how-to-install-pip-in-centos-7
3. Install nltk using pip3
   pip3 install nltk
4. Download nltk data to a Hadoop folder
   sudo python3 –m nltk.downloader –d /user/nltk_data all

Docker and Hadoop commands

1. docker exec –it 2FSDUS3 bash
2. docker run --hostname=quickstart.cloudera --privileged=true -t -i -v localpath:/src –publish-all=true -p 8888 cloudera/quickstart /usr/bin/docker-quickstart
3. hadoop fs -put twitter_data.txt /user/ravi/MR/input
4. hadoop jar /usr/lib/hadoop-mapreduce/hadoop-streaming-2.6.0-cdh5.7.0.jar \
   -file /src/mapper_count.py -mapper /src/mapper_count.py \
   -file /src/reducer_count.py -reducer /src/reducer_count.py \
   -input /user/ravi/MR/input/twitter_data.txt -output /user/ravi/MR/output

5. hdfs dfs -get /user/ravi/MR/output/ /src/

Folder Structure

- UbitLab2
  - Report.pdf
  - Video.mp4

- part1
  - Code
    - Commoncrawl
    - NYT
    - Twitter
  - Data
    - Commoncrawl
    - NYT
    - Twitter
- part2
  - Data
  - Output
- part3
  - Commoncrawl
    - Code
      - Data
      - Output
    - Images
  - NYT
    - Code
      - Data
      - Output
    - Images
  - Twitter
    - Code
      - Data
      - Output
    - Images
- Webpage

References

https://www.michael-noll.com/