Data Intensive Computing CSE 587- LAB 2

Apurba Mahanta-50288705
Sagnik Ghosh-50289151
University at Buffalo, The State University of New York
apurbama@buffalo.edu,sagnikgh@buffalo.edu

Abstract

The goal of this project is to demonstrate Data Aggregation, Big data analysis and Visualization on big data which is collected over multiple sources like Twitter,New York Times and Common crawl data using the API's offered by them.The outcomes of this project include evaluating the reliability of data form different sources like social media and news media.We worked on Hadoop 2.X and HDFS platforms and applied Big data algorithms like word count and word co occurence .We used Python as our data processing language.We used Tableau for visualization and evaluate the results obtained from the above infrastructure and algorithms.

1 Data Analysis on Gun violence and control in the United States

We chose our topic as Gun violence and control in the United States.Our subtopic are as follows:
a)Gunman
b)Gun Control
c)NRA
d)Gun Sense
e)Firearms and Second Amendment
Mass shooting and shooting at public places has been in the news in 2019.A lot of people and prominent figures have been expressing their views and concerns in various places like Twitter.A lot of criticism has also been seen in the public and hence this makes an interesting topic to understand the sentiments of people by collecting data from various sources and find the statistics about what people are thinking.

2 Big Data Infrastructure setup

1)We used the virtual image provided by our TA's to set up the environment.
2)We uploded the image in virtual box and provided the ram memory size as 8 gb
3)We installed ubuntu by running the image.
3)After the infrastructure is set up , we ran sample commands to test the framework like start-dfs.sh and hadoop commands like hadoop jar /home/cse587/hadoop-3.1.2/share/hadoop/mapreduce/hadoop-mapreduce-examples-3.1.2.jar wordcount /test/exam.txt /test/output.

3 Dataset size

Time Period : Jan 2019 to Apr 2019
1) Number of Tweets : 48390
2) New york times articles : 757
3) Commoncrawl data articles : 722

4 Implementation

We collected data from 3 different sources Twitter,NYT and Commoncrawl.We then cleaned the data,removed the stop words from the articles using NLTK word lemmatizer in the mapper process.The reducer process gave us the results which includes top 10 most frequent words and their co-ocurences.The implementation flow of the data is as shown below:

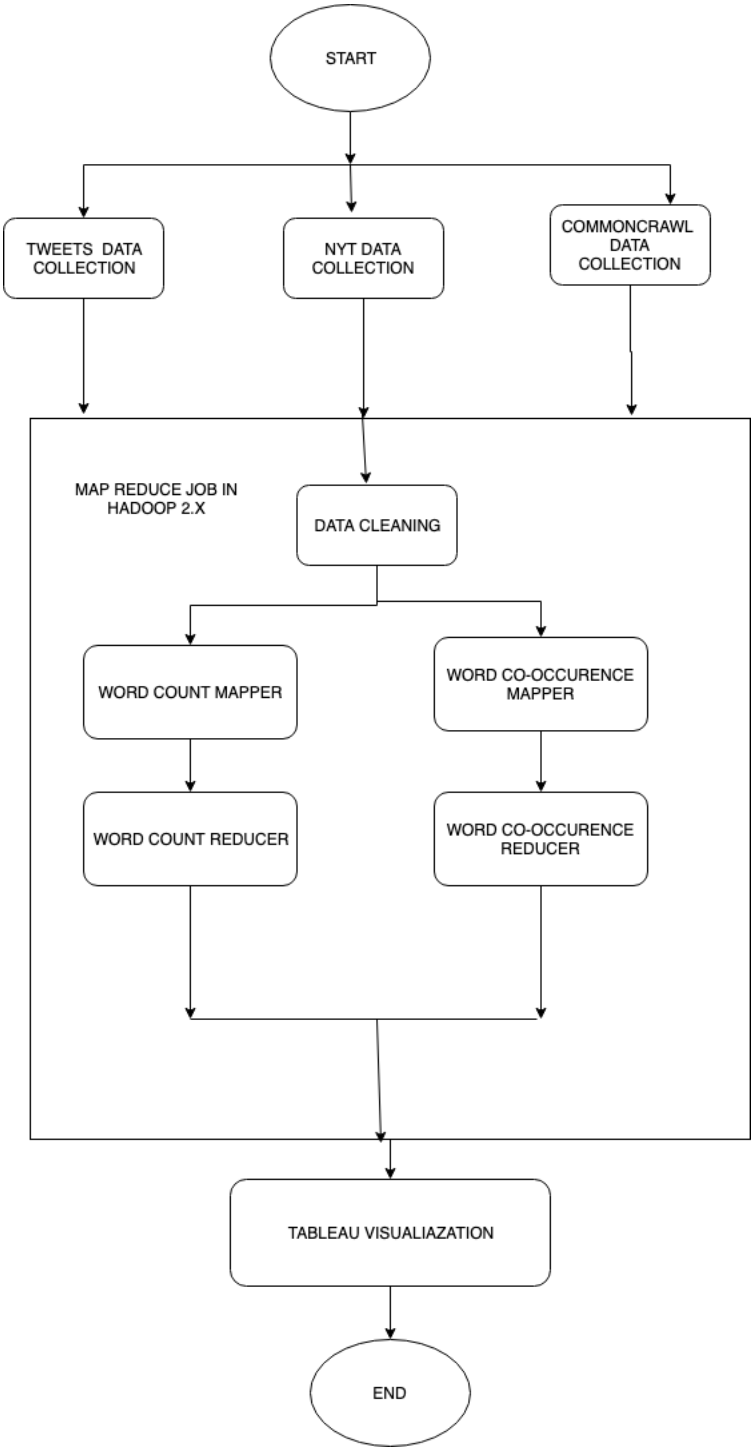

Figure 1: Implementation Flow

4.1 Twitter Data Collection

We used Tweepy api in python to collect data from twitter data.The script parses the tweets and collects text and user id.We removed the re tweets and duplicated tweets and saved the tweets to be used for map reduce jobs.The following hashtags was used for tweets collection:
#gunman
#shooting
#guncontrol
#gunsense
#firearms
#secondamendment
#nra
#gunlaws

4.2 NYT Data Collection

To gather New york times articles we used the api provided by NYT to collect the articles.After getting the related html contents from the url's returned,we then used Beautiful soup api to scrape and remove all the unneccesary tags in the html content .The api helped in getting the text from the body in the html content.Each article is then merged eventually to be further processed by map reduce framework. The following queries were used :
"gun control","gunman","nra","gun sense","firearms","second amendment"

4.3 Commoncrawl Data Collection

For collecting data from common crawl we used python to search the relevant domains in common crawl archive directory.The following domains were used :
guncontrolDomain = 'gun-control.procon.org/*'
guncontrolDomain = 'freerepublic.com'
gunsenseDomain = 'gunviolencearchive.org/*'
nraDomain = 'nraila.org/articles/*'
firearmsDomain = 'usatoday.com/story/*'
The indices searched were "2019-04","2019-09","2019-11","2019-14","2019-16"

After iterating over all the hits in the archive we get the warc files returned .We then unzip them on the fly and get the html contents.We then parse the html contents using Beautiful soup to get the text from the body of the html.We then have a check if the text is relevant to our keywords,we then store it in our disk .The relevant text is later merged and forwarded to map reduce framework for further processing

4.4 Word Count using map reduce framework

MapReduce is a processing technique and a program model for distributed computing based on java. The MapReduce algorithm contains two important tasks, namely Map and Reduce. Map takes a set of data and converts it into another set of data, where individual elements are broken down into tuples (key/value pairs). Secondly, reduce task, which takes the output from a map as an input and combines those data tuples into a smaller set of tuples

4.4.1 Mapper word count

1)We cleaned the data inside the mapper function using NLTK word lemmatizer python library.This removed all the stop words to make the data more meaningful.The mapper then emits a key value pair for each word and its count as 1.

4.4.2 Reducer word count

The reducer collects all the data and then calculates the word count for each word .We then sort all the data based on count and emit only the top 10 most frequent words from the dataset.

4.4.3 Mapper word co-occurence

1)We cleaned the data inside the mapper function using NLTK word lemmatizer python library.This removed all the stop words to make the data more meaningful.The mapper then check each word and its neighbour if they are among the top 10 words and emits the pair of it and count as 1

4.4.4 Reducer word co-occurence

The reducer collects all the data and then calculates the count for each word pair co-occurence .We then emit the total count for each word co-occurence.

4.5 Visualization using Tableau

To demonstrate the results we used Tableau as our visualization tool for word cloud visualization.There are in total 12 visualization results provided .The word cloud visualizations include the following results:
Twitter
1)Word count -Top 10 words - All data
2)Word co-occurence of the top 10 words-All data
3)Word count -Top 10 words - 1 day data
4)Word co-occurence of the top 10 words - 1 day data
NYT data
1)Word count -Top 10 words - All data
2)Word co-occurence of the top 10 words-All data
3)Word count -Top 10 words - 1 day data
4)Word co-occurence of the top 10 words - 1 day data
Commoncrawl data
1)Word count -Top 10 words - All data
2)Word co-occurence of the top 10 words-All data
3)Word count -Top 10 words - 1 day data
4)Word co-occurence of the top 10 words - 1 day data

5 Results and Visualization

These results have been published online and can be view by the following clickable hyper link:

https://public.tableau.com/profile/sagnik.ghosh#!/vizhome/CSE587-DataIntensiveComputing-Lab2/Main-Dashboard?publish=yes

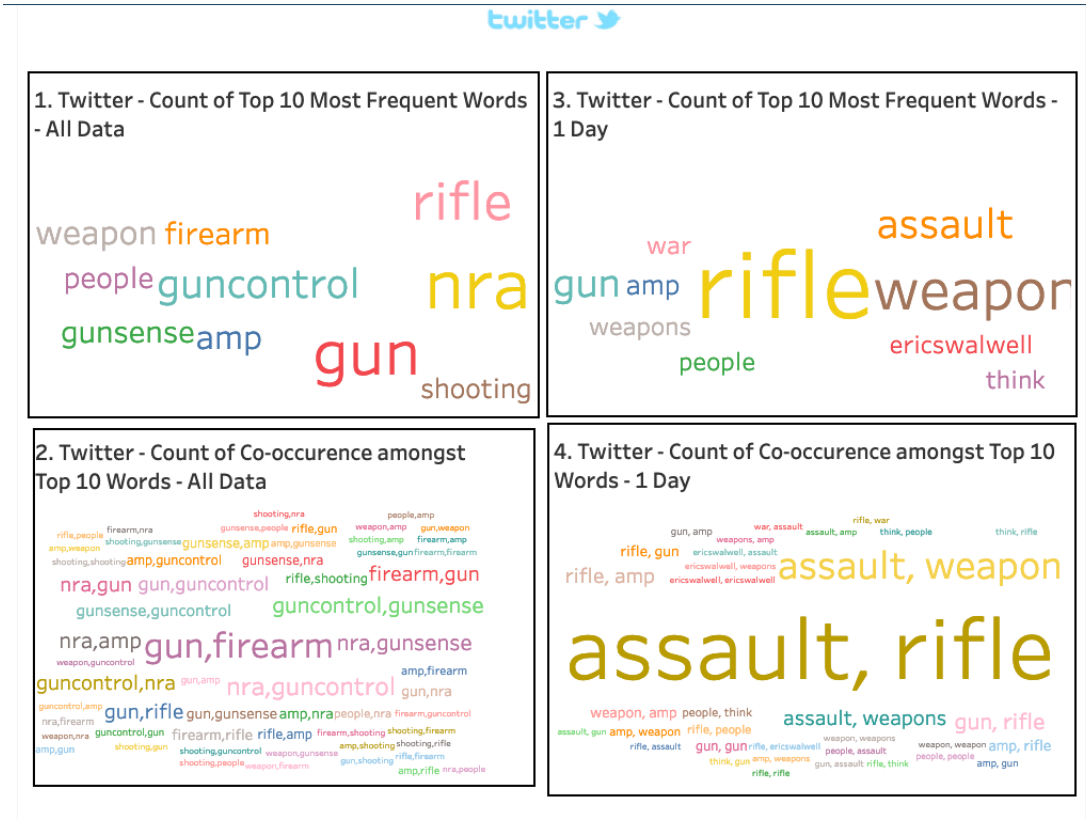

Figure 2: Twitter Word Cloud Visualization

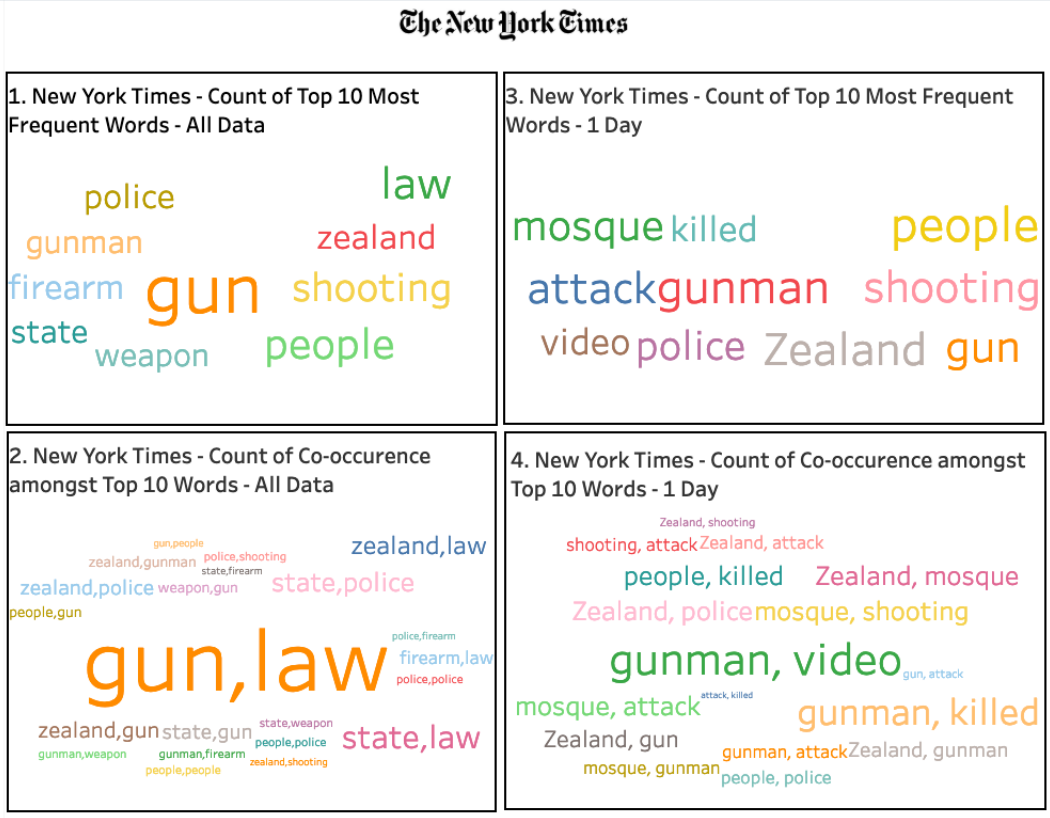

Figure 3: New York Times Cloud Visualization

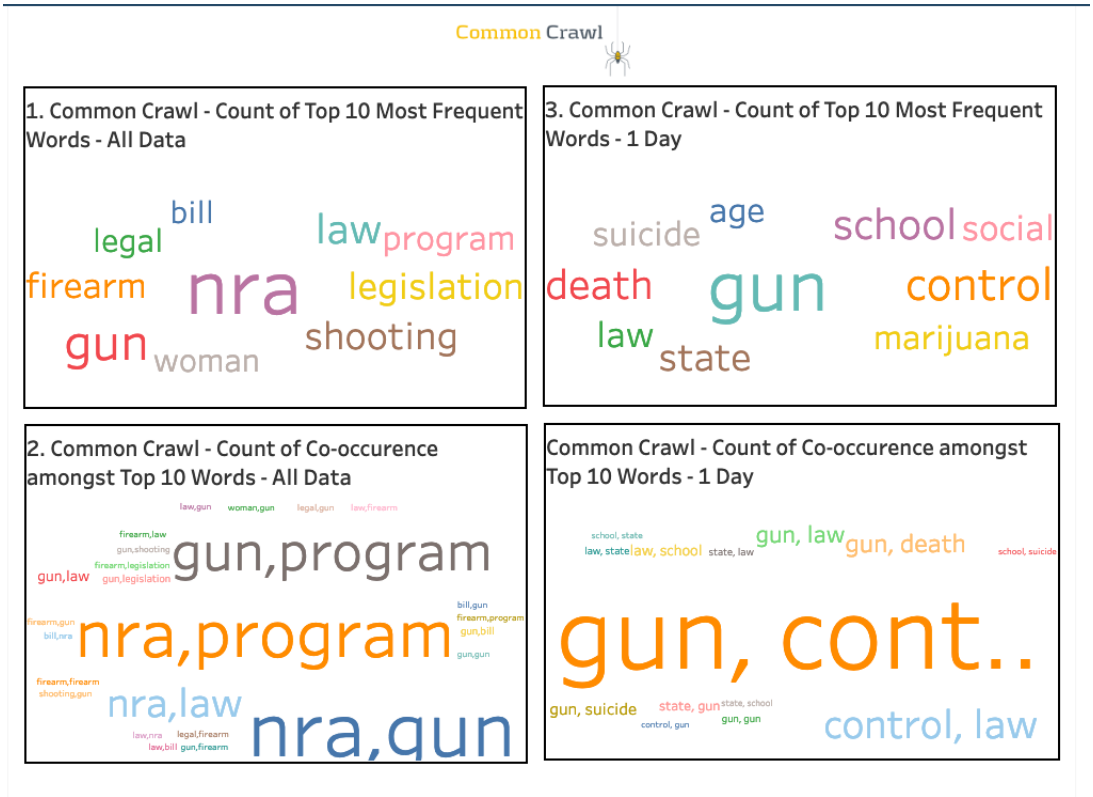

Figure 4: Commoncrawl word Cloud Visualization

6 Commands to execute

1) start-dfs.sh
2) hdfs dfs -put /part1/data/nyt/guns_nyt_all.txt
3)hadoop jar hadoop-3.1.2/share/hadoop/tools/lib/hadoop-streaming-3.1.2.jar -file /home/cse587/mapper_nltk.py -mapper 'python3 mapper_nltk.py' -file /home/cse587/reducer_nltk.py -reducer 'python3 reducer_nltk.py' -file /home/cse587/nltkandyaml.mod -input /user/apurbama/MR/input/twitter/guns_twitter_All.txt -output /user/apurbama/MR/output/twitter/wc/alldata/guns_all

4)hdfs dfs -getmerge /user/apurbama/MR/output/twitter/wc/alldata/guns_all guns_output.txt

References

1)https://cse.buffalo.edu/ bina/cse487/spring2019/Lectures/Lab2/
2) https://www.michael-noll.com/tutorials/writing-an-hadoop-mapreduce-program-in-python/
3) https://www.bellingcat.com/resources/2015/08/13/using-python-to-mine-common-crawl/
4) https://kb.tableau.com/articles/howto/creating-a-word-cloud