CSE 587:DIC Lab-2

Project Documentation

Abhinav Neelakantan
aneelaka@buffalo.edu

Yasha Ballal
yashaash@buffalo.edu

Overview:

We collected data from three sources- Twitter, New York Times (using NYT API) and CommonCrawl. We used amazon AWS S3 for storage and EMR for performing MapReduce. We performed word count and word co occurance on all the data collected and made a word cloud of top10 words that occur in each case.
Our topic is-Comics
Sub-topics- Marvel, DC, Avengers, Anime, Movies

Steps of execution:

1. Gather data

- Python was used to gather the data
- For twitter we used the twython package to collect tweets
- The data is first cleaned of all non alpha numeric characters and then stored as text files.
- For twitter 20000+ tweets are collected
- For NYT around 500 articles are collected
- For commoncrawl 500+ links are crawled
- For NYT we use the available packages in python and beautiful soup to crawl the data
- With Commoncrawl also, we finally use beautiful soup to get data
- The requests library is used to get and post from and to http/s links
-

2. Load data to S3 bucket for use in MR

- We create S3 bucket in AWS and load it with data to access for EMR
- We create a key value pair in EC2 before loading the data onto S3 as it is a prerequisite and makes the process easier

3. **Perform MapReduce on EMR**

- We perform word word count and word co occurrence on the data using MR
- We save the top 10 words in terms of word count and then perform word co occurance for them with respect to each other
- We used S3 buckets as our data source
- Output form the MR is again stored in form of txt files

4. **Plot graphs using D3**

- We run d3js on angular to produce interactive graphs for our word count and word co occurrence results
- Graphs are in the form of word clouds for all three data sets
- We use angular JS on top of NodeJS to create the d3 visualizations

Important Concepts:

- **Data Gathering**

1. Data is not always available in the format needed.
2. It has to be downloaded to local system and then changed to the proper format for operations
3. It is also necessary to check credibility of the data source

- **Data Cleaning**

1. Data is always contaminated with non alphanumeric characters that don't mostly play a role in analysis
2. Data can also be freed of stop words, stemmed, lemmatized if necessary

- **Setting up AWS**

  1. Its important to set up key value pairs and security keys and data buckets before beginning MR operations

- **MapReduce**

1. MR has two primary functions: Mapper and Reducer.
2. There are two complimentary functions: Combiner and Partitioner
3. We can also perform operations other than Mapping and Reducing like data cleaning using the MR module

- **Plotting on d3**

1. d3 is a javascript library to make dynamic visualizations.
2. It can be run independently or with Angular or React on top of NodeJS.

Process:

- **Data Gathering:**

1. **Twitter:**
    - We used Twython library to download tweets
    - we first establish credentials using twitter API keys
    - We extract tweets based on our topics and make sure they are unique by disabling retweets and repeats
    - We then remove the non alphanumeric characters using regex to make it plain text
    - We then flush the contents to a text file in the end

2 .**New York Times**

- We created a NYT account to access the API
- We used requests library to get web pages
- We then use beautiful soup, run the html parser through it and extract the <p> tags using the <a> tags

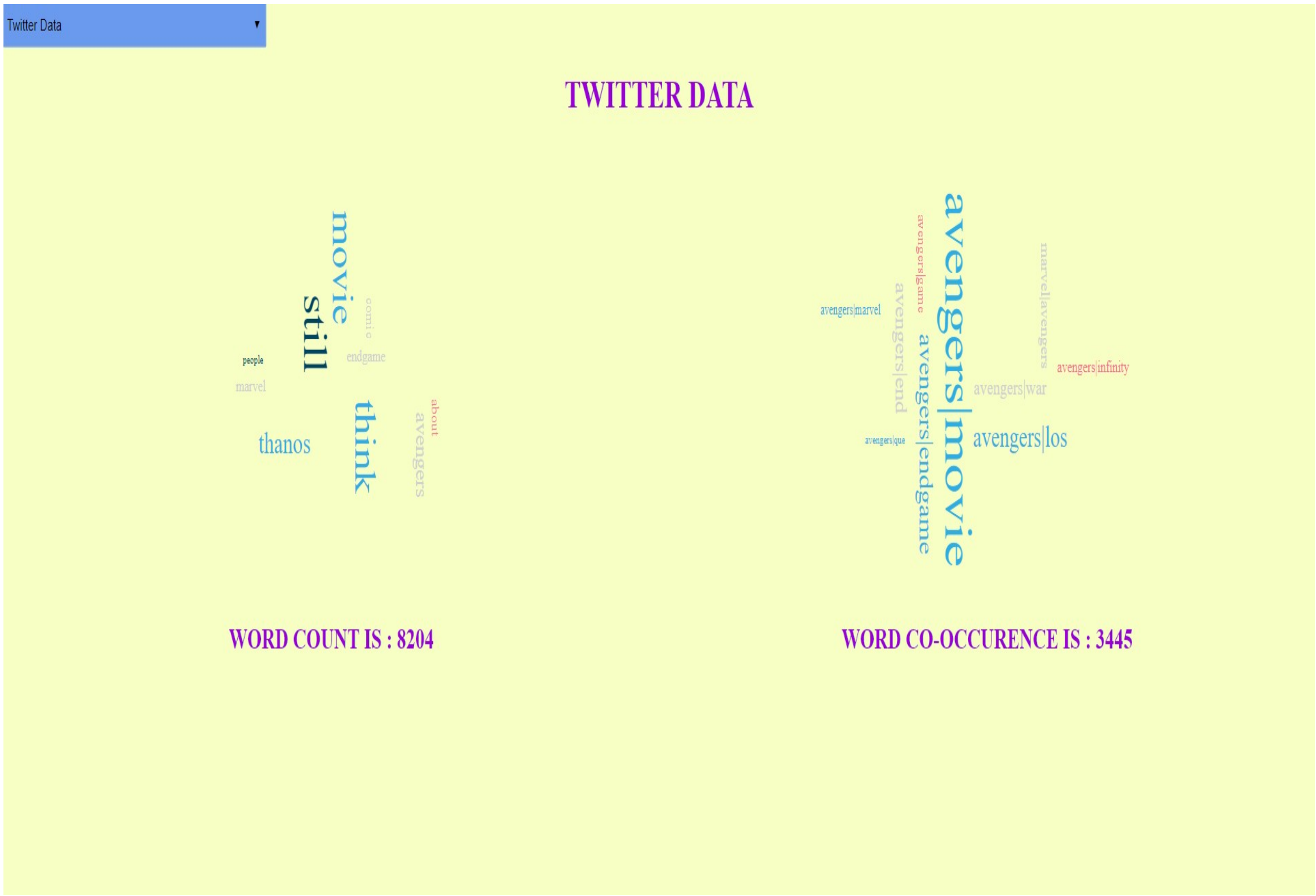
3.**Common Crawl**

- We used a script that when provided with the month and domain name, extracts data from the commoncrawl index
- What we get using the index is a gzip warc file that contains the contents we need
- We unzip the gzip file and read the warc daa contents
- We use beautiful soup as we did in the NYT to get the p contents
- We use only the first gzip file from the list of files in the json received because for a month, most often its the same content repeated over and over.
- We also skip the process if length of the file is less than 500 because lengths less than 500 usually result in 301(page moved permanently) or 404 (page not found)

- **MapReduce**
- We performed word count and word co occurance using EMR on AWS
- We picked up the top 10 words that occurred with each data set to make out plots in the next step of the process
- We got a EMR cluster working and fed it data(twitter, NYT and common crawl) set up in an S3 bucket.
- The MR was performed using python

- **D3 visualization**
- We used the d3 library with angular JS on top of NodeJS for our visualizations
- We plot the top 10 words in terms of count and co occurence between them as word cloud
- We plot 2 graphs each for all three data sources i.e. Twitter data, NYT and common crawl

WORD CLOUDS FOR TWITTER



TWITTER DATA

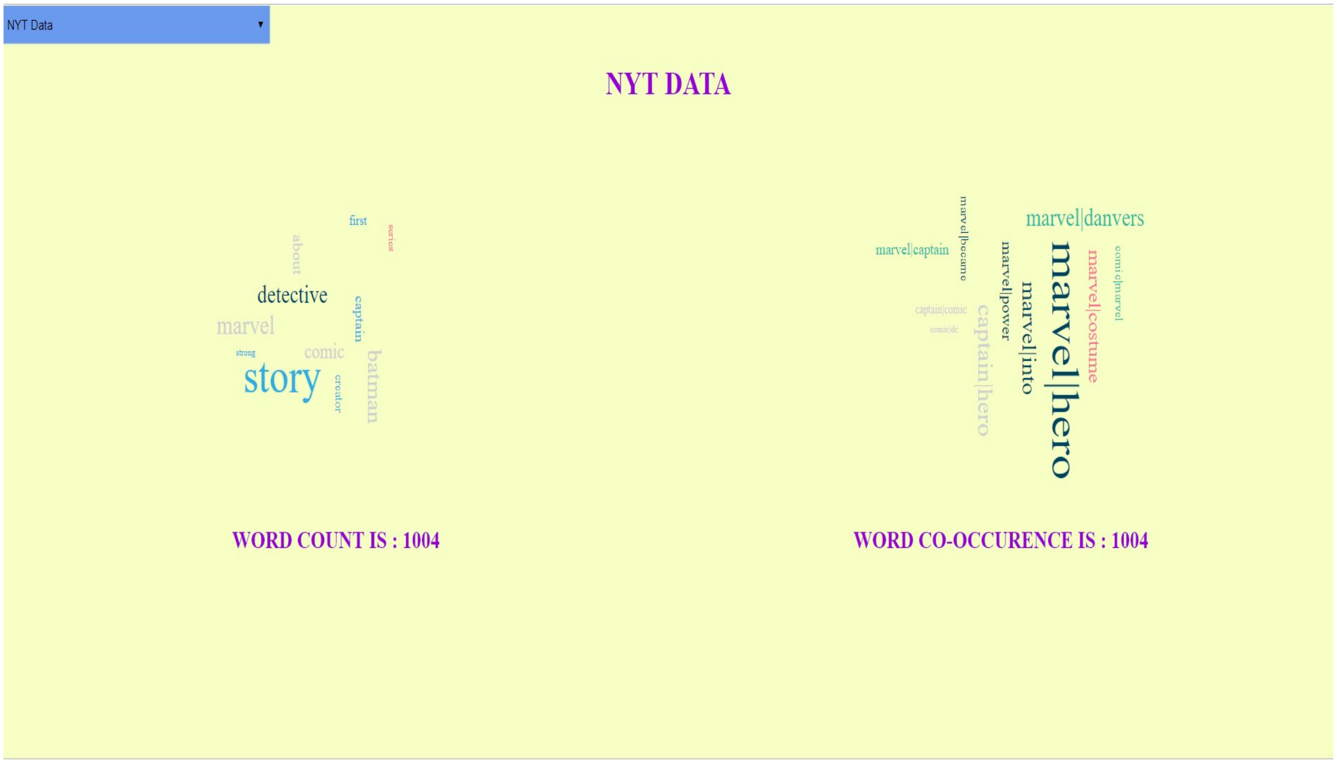
WORD COUNT IS : 8204

WORD CO-OCCURENCE IS : 3445

Analysis:

- Twitter is very current and gives a very good sense of what the general news is about in the current world. With Avengers Endgame releasing this week, most of the references with respect to comics also goes to avengers.

WORD CLOUDS FOR NEW YORK TIMES



NYT DATA

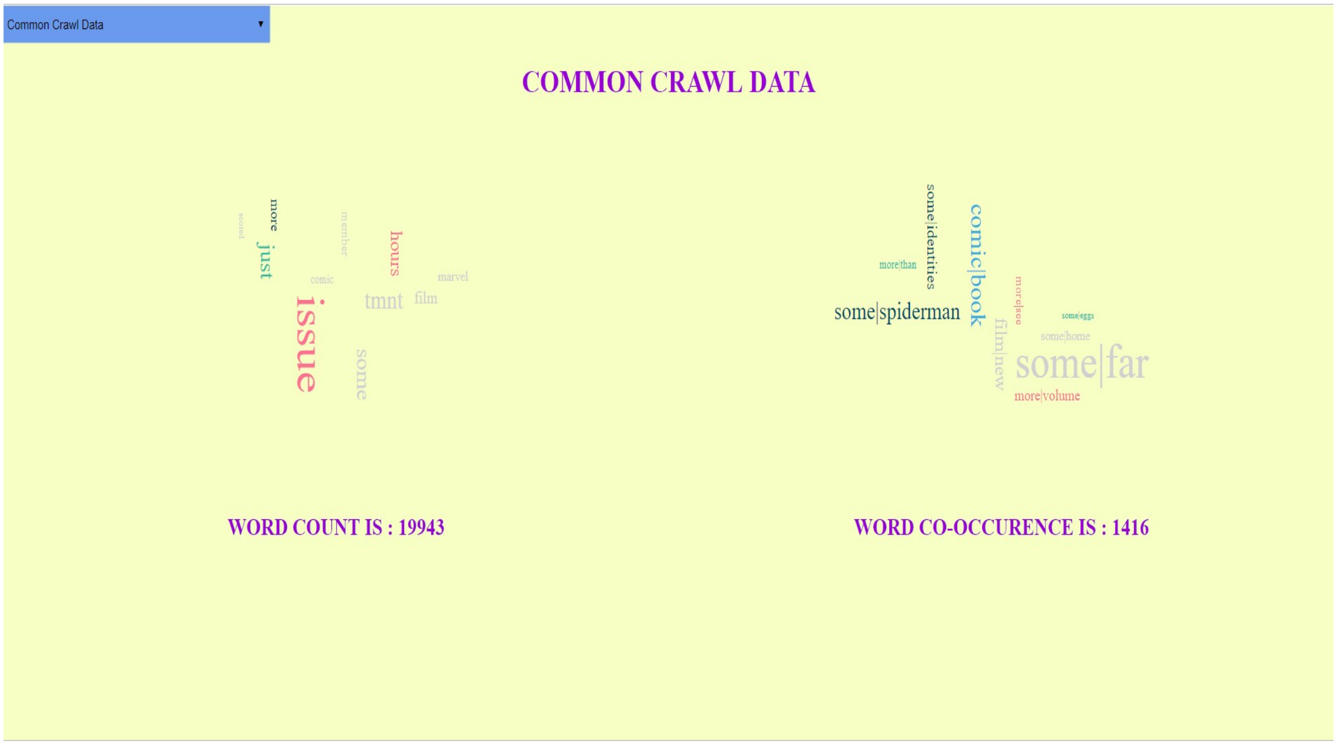
WORD COUNT IS : 1004

WORD CO-OCCURENCE IS : 1004

Analysis:

- The NYT data is more balanced in terms of the word count but marvel really steals the show in terms of co occurrence again owing to the impending avengers release
- Even the newspaper is focusing a lot of stories and content on marvel because of the release of Captain Marvel and Avengers Endgame back to back.

WORD CLOUD FOR COMMON CRAWL



Analysis:

- Common crawl seems to not be associated with a narrow source of data or current news but seems more to reflect the overall sentiment of the entirety of the internet.
- It has a more general outlook to things as comic book and spiderman seem to be the only direct references made while the other are either indirect or implied

References:

- https://gist.github.com/Smerity/56bc6f21a8adec920ebf
- https://stackoverflow.com/questions/35938188/twitter-api-how-to-exclude-retweets-when-searching-tweets-using-twython
- https://stackoverflow.com/questions/22520932/python-remove-all-non-alphabet-chars-from-string/22521235