CSE 4/587 Data-Intensive Computing

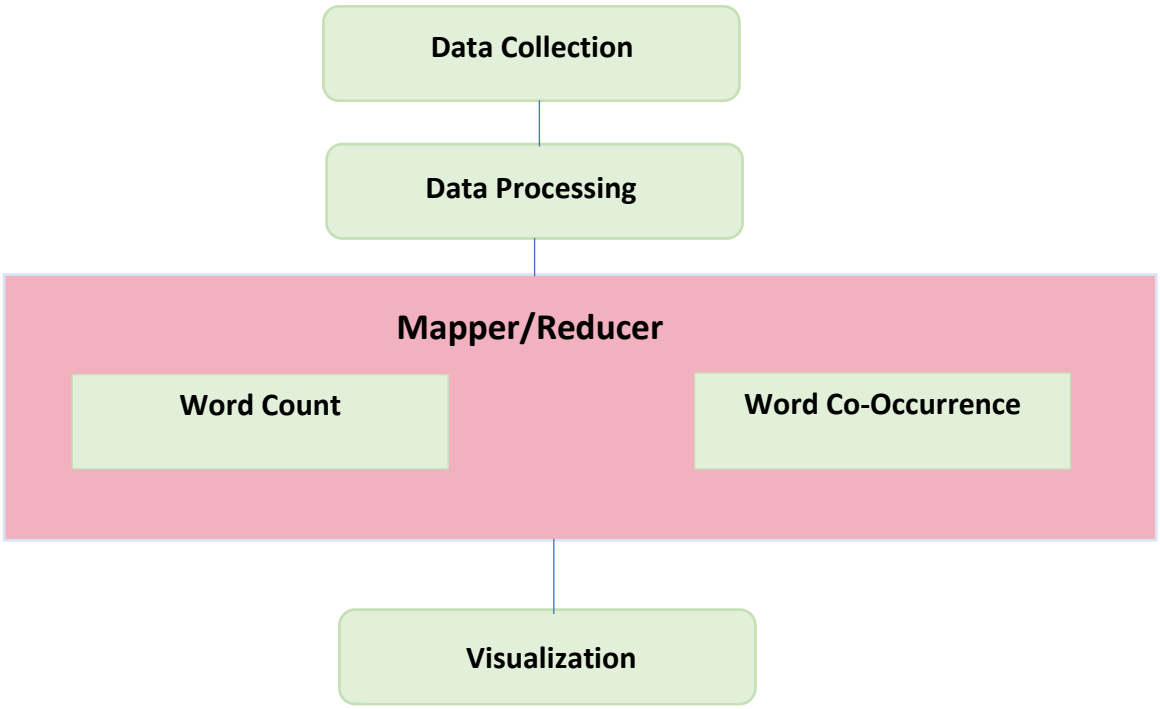
LAB2: DATA AGGREGATION, BIG DATA ANALYSIS AND VISUALIZATION

Sonali Yeshwant Naidu (50299652)
Pratibhaa Reddy Kandimalla (50299669)

Introduction:

In this lab we worked on Data Aggregation, Big data analysis and Data Visualization. The topic chosen for this Data Analysis is 'Sports'.

Below is the block diagram showing the different steps involved.



Part 1

Data Collection:

- Collected data from three different sources such as "Twitter", "NY Times" and "Common Crawl".
- We have extracted the data related to topic "Sports" also using the subtopic/keywords for better data collection such as Basketball, Football, Golf, Badminton, Tennis, Cricket.
- The above data was collected using APIs for twitter and NY Times.

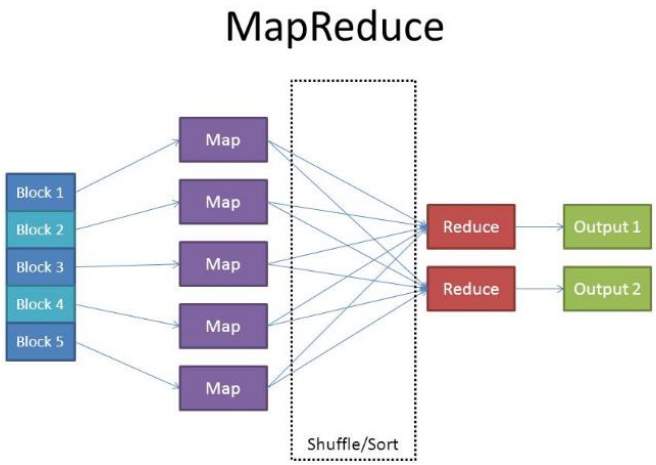
Data Processing:

- The data extracted from the above sources i.e. Twitter and NY Times was cleaned for any duplicates, stopwords, unwanted charactes like punctuation marks, special characters etc.
- Downloaded and used NLTK in python to preprocess the same.

Mapper/Reducer:

- Once HDFS and Virtual Machine were setup the WordCount and Word Co-Occurrence was run on the above processed data.
- On successful execution of Mapper/Reducer programs, the words along with their count was their output.

- We ran the word cooccurence program on the same data collected.



Data Visualization:

- The output files of Mapper/Reducer for the WordCount and the Word Co-occurrence is loaded into Tableau for data visualization.
- Data processed is visualised in Tableau in the WordCloud format.

Project Extension:

The same project can be used on any other data sets for Data Collection, Analysis and Visualization. The data could be from any varied sources,which can be preprocessed and further analyzed. This data can be visualized using Tableau or any other data visualization tool.

Part 2

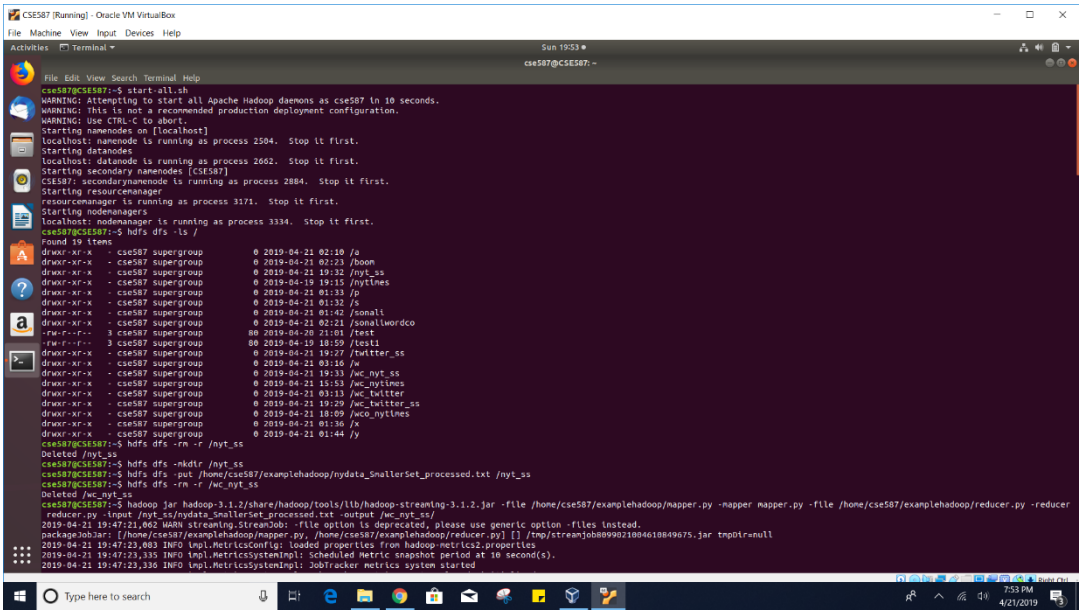
We used the VM Box to setup the Hadoop Infratructure for storing the Big Data collected from Twitter, NYTimes and Common Crawl. The mapper/reducer was run to clean and parse the data sets into words, remove stop words, stem the words (ex: running to run) and reduce will count the useful words.

Word Count: This Programs is run on the data collected to parse the words and get the count of each word in the data which is later used for data visualization to understand the most commonly used words from Twitter, NYTimes and CommonCrawl.
Word Co-Occurrence: This Program was run to find the word co-occurrences in the collected data.

Below is the screenshot from VM Box with the hdfs directory and certain commands for running the Mapper/Reducer in it.
Few of the commands used were:

- hdfs dfs -ls / to view the list of directories

- hadoop jar hadoop-3.1.2/share/hadoop/tools/lib/hadoop-streaming-3.1.2.jar -file /home/cse587/hduser/mapper.py -mapper mapper.py -file /home/cse587/hduser/reducer.py -reducer reducer.py -input /nytimes/nydata.txt -output /wc_nyt/
-



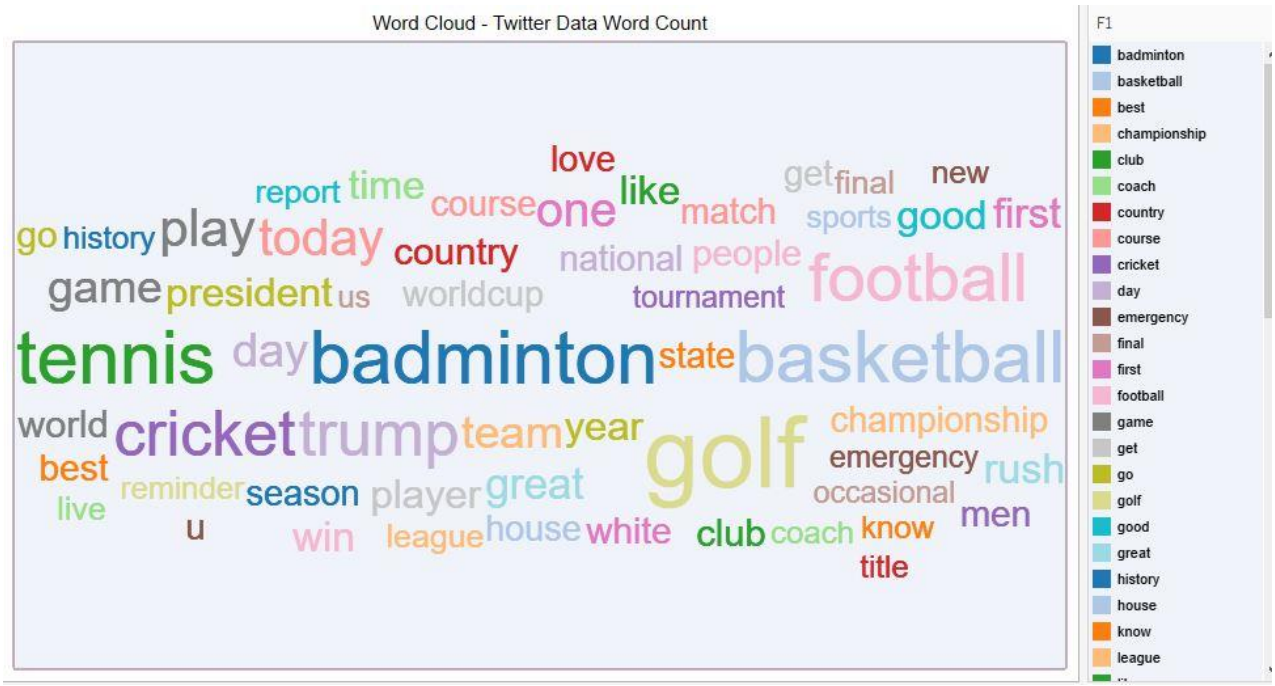
Part 3

The mapper/reducer and word co occurrence code was run on all the three sets of data and visualized using Tableau.

Below are the screenshots of the code visualized for the word count and word co ocurences data.

Word Clouds:

Word Count:
Wordcount on Twitter:

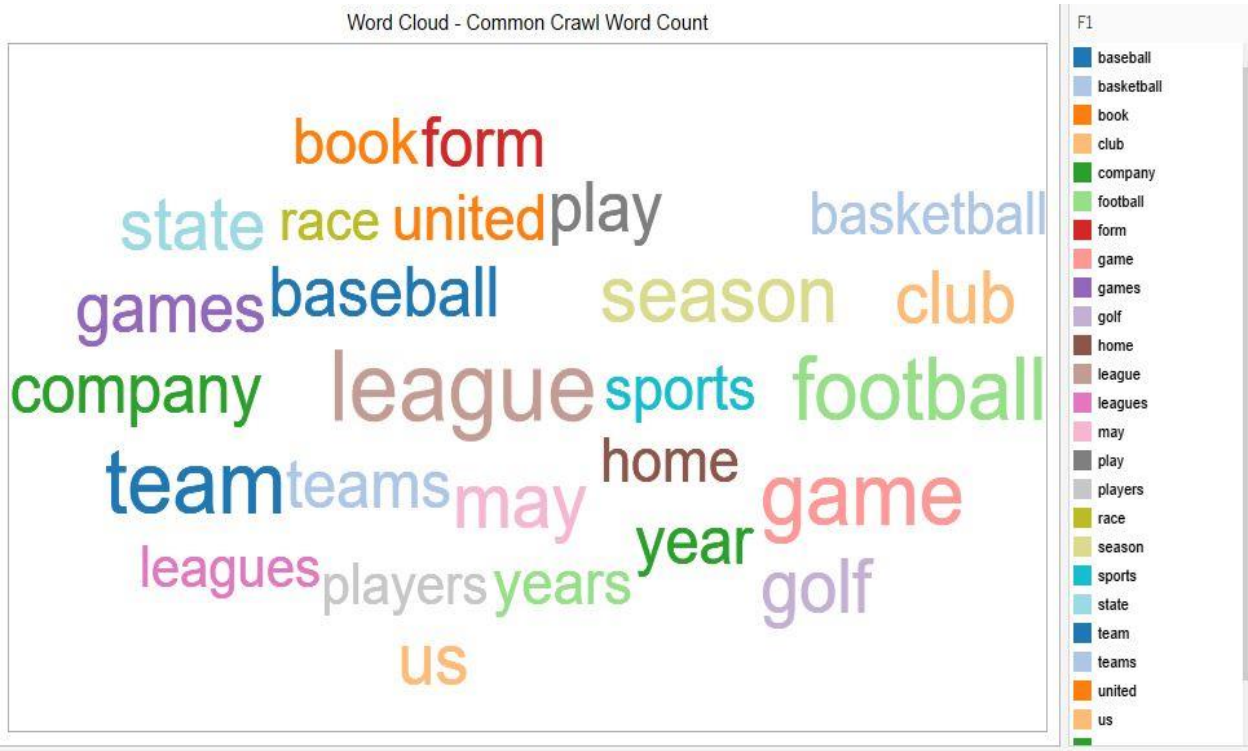


Word Cound on smaller set of Twitter data:

Wordcount on NYTimes:

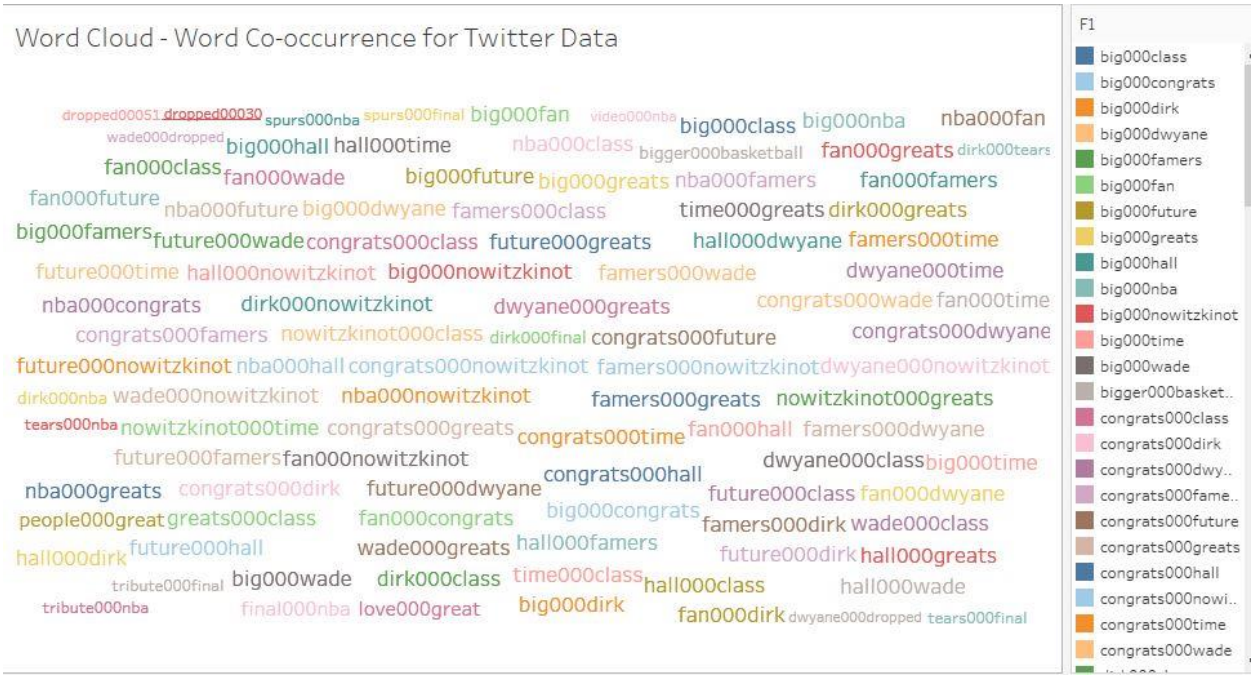
Word Cound on smaller set of NYTimes data:

Wordcount on Common Crawl:

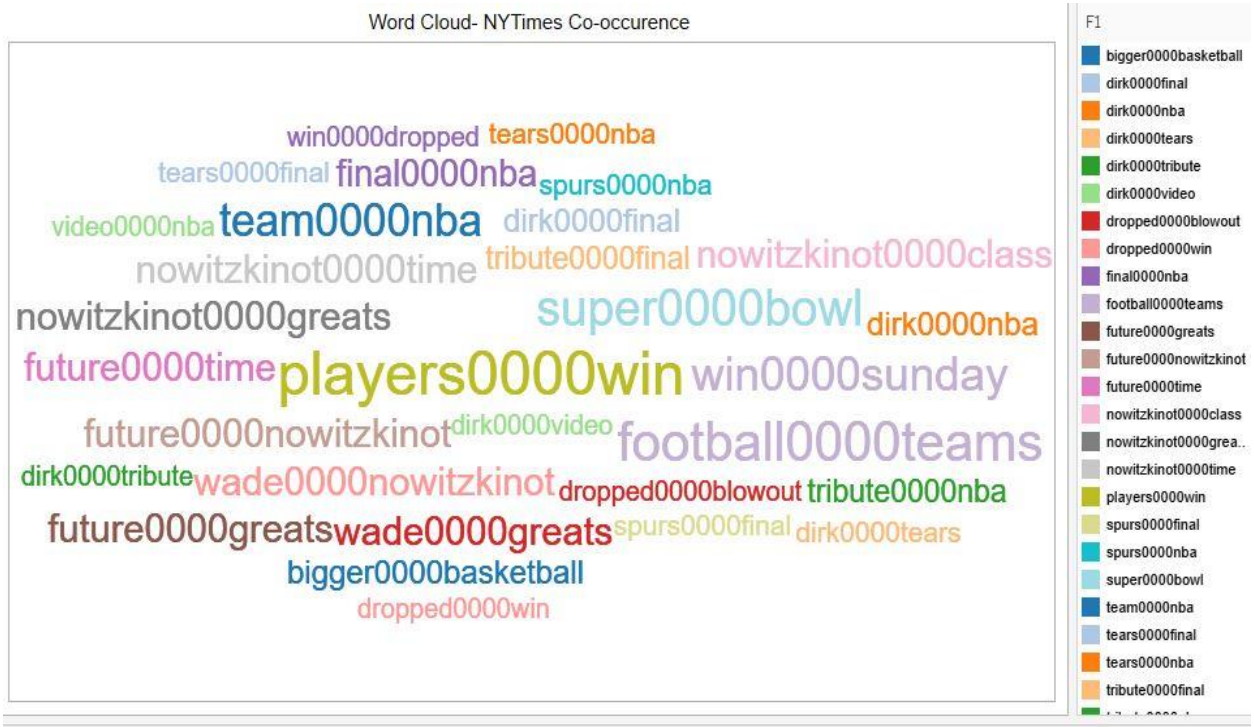


Word Co-Occurrence:

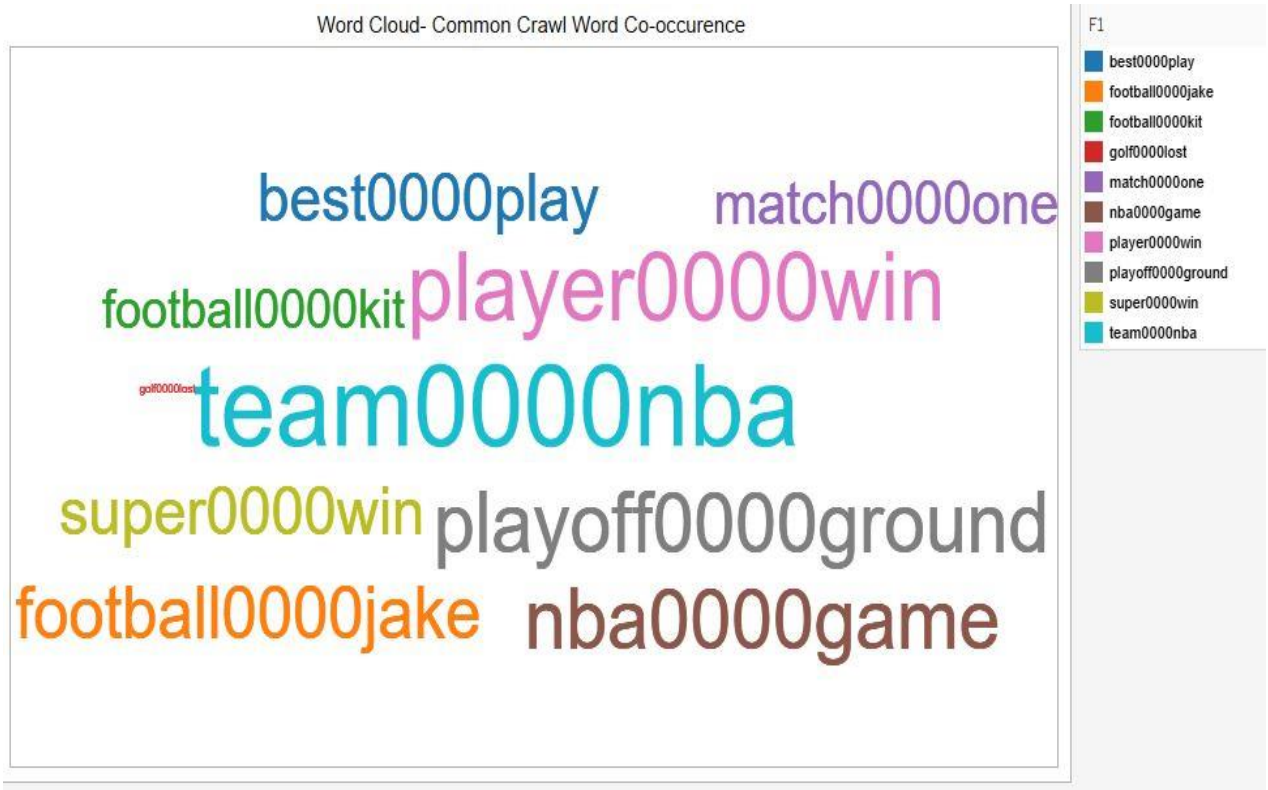
Word Co-occurence on Twitter data:



Word Co-occurence on top 30 NYTimes data:



Word Co-occurence on top 10 Common Crawler data:



Directory Structure:

sonaliyeLab2.zip

- **Report.pdf**
- **Video.mp4**
- **part1 (folder)**
  - **Code** *included all the scripts related to data collection for Common Crawl, Twitter and NYT*
  - **Data**
    - Twitter -> Twitter Data, Twitter Data Processed, Twitter WC and Twitter Co-occurence
    - NYT -> NYTimes Data, NYTimes Data Processed, NYTimes and NYTimes Co-occurence
    - Commoncrawl -> Common Crawl Data, Common Crawl and Common Crawl Co-occurence
- **part2 (folder)**
  - *mapper.py reducer.py, pairsmapper, dataprocessing*
  - *other files (screenshots/results)*
- **part3 (folder)**
  - **Twitter** (folder)
    - Code (folder) -> *mapper.py reducer.py, pairsmapper, dataprocessing*
    - Images (folder) -> *all visualizations .jpg/.png AND .js /.twbx files*
  - **NYT** (folder)
    - Code (folder) -> *mapper.py reducer.py, pairsmapper, dataprocessing*
    - Images (folder) -> *all visualizations .jpg/.png AND .js /.twbx files to create NYT images*
  - **Commoncrawl** (folder)
    - Code (folder) -> *mapper.py reducer.py, pairsmapper, dataprocessing*
    - Images (folder) -> *all visualizations .jpg/.png AND .js /.twbx files to create CC images*

References:

NYTimes: https://www.youtube.com/watch?v=zXif_9RVadI
CommonCrawl: https://rushter.com/blog/python-fast-html-parser/