Hadoop instructions:

1. Extract the tweets or new york times data in txt files and store in a directory eg. tweets.

2. Now first start hadoop by typing the start-hadoop.sh command in the terminal.

3. Once hadoop is started place your mapper and reducer python files in the $HOME directory.

4. Now type "hdfs dfs -put $HOME/tweets input" in the terminal. This will create an internal directory named input which is accessible to hadoop.

5. Once this command runs its time to run hadoop for calculating the word count.

6. For doing that type the following command:

```
hadoop jar /home/hadoop/hadoop/share/hadoop/tools/lib/hadoop-streaming-2.6.4.jar -file mapper.py -file reducer.py -mapper mapper.py -reducer reducer.py -input input -output output
```

7. Now the counts will be calculated and stored in the output directory.

8. Now move the output directory to any local folder for further processing using the following command:

```
hdfs dfs -get output $HOME/nyoutput
```

Hadoop Output Processing

Once the output is generated each line in the output file is of the form
<word> <count>

1. We first convert this into a reverse form i.e. "<count> <word>" using the following unix command:
   sed -e "s/\([a-z0-9\'\._\-\=\\\/\:*\(\)\+\@\&\~]\+\)\t\([0-9]\+\)/\2 \1/g" part-00000 >sorted.txt

2. Once this is done we sort the file in descending order based on the counts using the following command:
   sort -n -r sorted.txt >final.txt

3. Now when we have the sorted file we convert it into json type incomplete format which we manually copy paste in .js file as a data to the variable which will be further used by the d3 code of ours.
   sed -e "s/\([0-9]\+\) \([a-z0-9\'\._\-\=\\\/\:*\(\)\+\@\&\~]\+\)/\{text\: \'\2\', size\: \'\1\'\}\,/g" final.txt >json.txt

D3 instructions

1. For D3 i have created 2 html files d3_tweet.html and d3_ny.html one for twitter data and one for nytimes data.
2. Just run the html file in your browser.
3. Each one takes data from their respective .js files which contains the word count data and co occurrence data in the JSON format.
4. Please make sure to keep the html files in their respective folders that I've created since the paths given are dependent on the folder structure

Submission Folder Structure :

A. Part 1:
This part contains the 3 notebooks of each Ch3, Ch4, Ch5

B. Part 2:

1. The following folder contains outputs from running hadoop command on all the one day, one week data from twitter and NYtimes:

   Input data:
   Hadoop_data->nytimes,
   Hadoop_data->nyoneday,
   Hadoop_data->tweets,
   Hadoop_data->tweetoneday

   Output data:

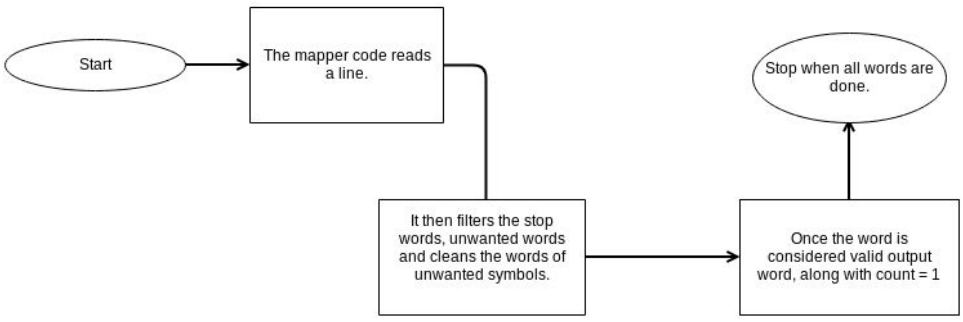
   Hadoop_data->nytimesoutput
   Hadoop_data->tweetoutput

2. The mapper.py is same for both the tweet data and the Nytimes data
   We've created 2 different cooccurence mappers for both types of data i.e. coMapper.py and cpMapperny.py

3. The d3Files folder contains my d3 html files and the required js files.

4. The instruction video is present at
https://buffalo.box.com/s/4327w9d6sp5s4wqsi9gl54i1ydm3ha2d

Mapper Block Diagram:



References:

1. https://github.com/wvengen/d3-wordcloud
2. https://en.wikipedia.org/wiki/MapReduce
3. https://blog.matthewrathbone.com/2016/02/09/python-tutorial.html