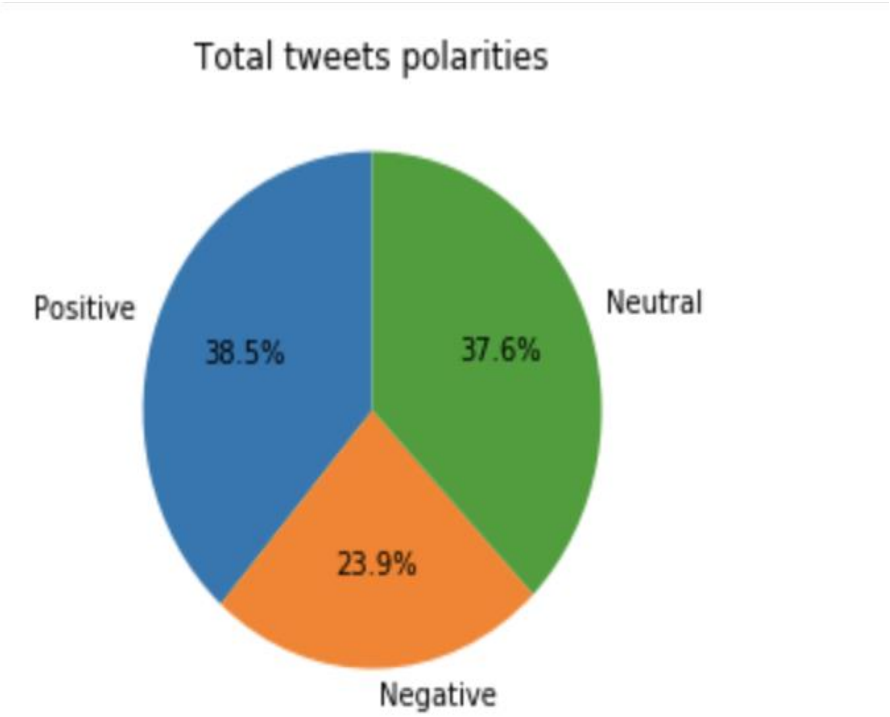
1. Sentiment Analysis:

- A twitter developer account was created initially using the tutorial 5 provided in the CSCI-5408 labs [1].
- A Twitter application is created, and keys & tokens are generated for getting access to extract the tweets from twitter.
- A total of 1000 tweets have been extracted from twitter using the tweepy searching API in python script [2]. The keywords “Canada”, “Canada import”, “Canada export”, “Canada vehicle sales”, “Canada Education” are used for finding the tweets.
- Tweets have been cleaned by removing the urls, special characters except ‘+’ and ‘-‘ as these characters are part of some positive and negative words.
- Bag of words for each tweet is created by counting the frequency of each word in that tweet.
- Positive and negative list of words were gathered from internet[7] and each word in the tweet was cross checked with the available positive and negative words.
- Each tweet is labelled with polarity as positive if more number of words are positive, tweet is labelled with polarity as negative if more number of words in the tweet are negative, tweet is labelled as neutral if positive and negative words are matched or if there are no positive and negative words in the tweet.
- A data frame is created in pandas with tweet number, tweet message, matched words and polarity of tweets.
- Dataframe output table is written to JSON file(**tweets_analysis.JSON**) which is provided in the attached submission zip folder.
- The logic used for extracting the tweets, cleaning the tweets, doing sentiment analysis is provided in **Sentiment_Analysis.ipynb** python script which is included in the submitted zip folder.
- The output table is present in the submitted python script and the same results were stored in output JSON file (**Sentiment_Analysis_output.JSON**). The sample result is as below:

	Tweet	Message	Match	Polarity
0	1	Meet Jessica Yaniv a Jewish of course trans ac...	[refused]	negative
1	2	Reminder that in Canada you can pander to peop...	[pander, bother, extremist, denounce]	negative
2	3	You know its hot in Canada when	[hot]	positive
3	4	This is the brave woman taking multiple female...	[brave, refusing]	neutral
4	5	The government of Canada is considering whethe...	[assault]	negative
5	6	hgmackinnon HastroTags Canadas Greece	[]	neutral
6	7	Scientific articles published 2016 1 US 25935 ...	[]	neutral
7	8	pklinkne VictorAsal You lucky folk get to go l...	[lucky]	positive
8	9	Dateline Lviv Ukraine Lviv radio station promo...	[myth]	negative
9	10	Theres nothing exemplary about a man who rapes...	[exemplary, fuck]	neutral
10	11	Good morning So what kind of Canada do you wan...	[]	neutral
11	12	The people at Ontario Proud who helped fordnat...	[helped]	positive
12	13	BillMorneau Comical Youre still looking at rai...	[]	neutral
13	14	If youre 18+ and a Canadian citizen on electio...	[]	neutral
14	15	Why has lraitt not tweeted in support of Andre...	[attack, stupid]	negative
15	16	You know its hot in Canada when	[hot]	positive
16	17	stonecold2050 Thats in Canada	[]	neutral
17	18	Suicide hotline numbers United Kingdom 116 123...	[dies, suicide]	negative
18	19	Many have been seeking her guidance on job and...	[guidance, works]	positive
19	20	Retro Throwback UMVC3 Season 4 PS4 Online Tour...	[]	neutral

Pie chart showing the polarities of all tweets:



2. Semantic Analysis:

- As there are 1578 articles in the two given sgm files, each of the article is stored in a separate file.
- A python script was written to parse the sgm files and find the <text> and </text> tags in the provided sgm files. The data between <text>, </text> tags was identified and written to a new text file using a python logic.
- The data in each article is cleaned by removing all the tags and special characters.
- Total of 1578 files generated from the two sgm files and are stored in a directory separating from rest of the file structure.
- All the articles which were created above are read one by one to do the frequency counts for words “Canada”, “Halifax”, “nova scotia” along with document frequency. The output as shown below:

	Search Query	Documents containing term(df)	Total_documents/df	Log(Total_documents/df)
0	Canada	27	1578/27	1.76674
1	Halifax	0	1578/0	NA
2	Nova Scotia	0	1578/0	NA

- The document/Article which has the highest occurrence of the word “Canada” is determined by doing the frequency count of word “Canada” in all the documents. The term frequencies of word “Canada” in all the documents is as below:

	term	document	total_words	frequency	f/m
2	canada	Article1124.txt	15	1	0.066667
85	canada	Article829.txt	186	2	0.010753
94	canada	Article1333.txt	221	1	0.004525
271	canada	Article1056.txt	18	1	0.055556
388	canada	Article352.txt	101	3	0.029703
501	canada	Article452.txt	179	3	0.016760
506	canada	Article335.txt	359	1	0.002786
522	canada	Article1542.txt	643	1	0.001555
552	canada	Article282.txt	175	1	0.005714
665	canada	Article520.txt	16	1	0.062500
687	canada	Article865.txt	424	1	0.002358
716	canada	Article37.txt	98	1	0.010204
764	canada	Article251.txt	27	1	0.037037
843	canada	Article1341.txt	112	2	0.017857
958	canada	Article706.txt	145	1	0.006897
976	canada	Article1005.txt	86	2	0.023256
991	canada	Article1004.txt	246	3	0.012195
1023	canada	Article511.txt	408	2	0.004902
1112	canada	Article500.txt	640	1	0.001563
1253	canada	Article411.txt	182	4	0.021978
1286	canada	Article1306.txt	78	1	0.012821
1308	canada	Article1338.txt	110	2	0.018182
1412	canada	Article1261.txt	133	2	0.015038
1414	canada	Article1513.txt	74	2	0.027027
1499	canada	Article763.txt	86	1	0.011628
1558	canada	Article986.txt	153	1	0.006536
1571	canada	Article574.txt	14	1	0.071429

Document with high frequency of word “Canada”

```
#displaying the article in which "canada" is present higher number of times

df.loc[df['frequency'] == df['frequency'].max()]
```

	term	document	total_words	frequency	f/m
1253	canada	Article411.txt	182	4	0.021978

```
#dsiplaying the content of article in which "canada" is present higher number of times

InputFile = open("./Articles/Article411.txt", "r")
print (InputFile.read())
```

```
gm canada workers far apart in talks union
toronto oct 20 the canadian unit of general motors corp
and the union representing its 40000 workers remain far apart
over local issues in contract talks two days from a threatened
strike a union spokesman said
the deepest divisions appeared to be between general motors
of canada ltd and the canadian auto workers local representing
17000 workers at an assembly plant in oshawa ontario union
spokesman wendy cuthbertson said
local 222 is miles apart said cuthbertson the local
assembles fullsize pickup trucks the buick regal and the
pontiac 6000
local issues there included shift schedules transfers and
working conditions union president bob white said
the union has threatened to strike at 1000 hrs edt 1400
gmt on thursday unless it has reached a tentative settlement
with the automaker by then
bargaining was scheduled to continue late into the night
tuesday in an effort to avert a walkout said cuthbertson
on monday the union accepted an economic offer from gm
canada that largely matched the payandbenefit pattern reached
earlier at chrysler and ford in canada
```

- The article with highest relative frequency count was determined by calculating “frequency of occurring Canada”/”Total number of words in the file”

```
#finding the article having highest relative frequency for word "canada"

df.loc[df['f/m'] == df['f/m'].max()]
```

	term	document	total_words	frequency	f/m
1571	canada	Article574.txt	14	1	0.071429

```
#dsiplayinh the article having highest relative frequency for word "canada"

InputFile = open("./Articles/Article574.txt", "r")
print (InputFile.read())
```

```
canada minister says g7 action has provided stability consultations continuing
blah blah blah
3
```

- The python script used for parsing the sgm files, creating separate files for each article and doing semantic analysis **(Semantic_Analysis.ipynb**) is provided in the submitted zip folder.

References:

[1] Dal.brightspace.com, 2019. [Online]. Available: https://dal.brightspace.com/d2l/le/content/98340/viewContent/1337724/View. [Accessed: 21- Jul- 2019].

[2]"Standard search API", Developer.twitter.com, 2019. [Online]. Available: https://developer.twitter.com/en/docs/tweets/search/api-reference/get-search-tweets.html. [Accessed: 21- Jul- 2019].

[3] Dal.brightspace.com, 2019. [Online]. Available: https://dal.brightspace.com/d2l/le/content/98340/viewContent/1350102/View. [Accessed: 22- Jul- 2019].

[4]"Natural Gas Exports Historical Summary by Term - Open Government Portal", Open.canada.ca, 2019. [Online]. Available: https://open.canada.ca/data/en/dataset/28754a8c-8734-44ae-a66b-0d7a5cfa0a0b. [Accessed: 22- Jul- 2019].

[5]"Second Language Immersion Schools in Canada - Open Government Portal", Open.canada.ca, 2019. [Online]. Available: https://open.canada.ca/data/en/dataset/2bfebd29-1a98-4c57-9134-93f1b18190ea. [Accessed: 22- Jul- 2019].

[6]"New motor vehicle sales, by type of vehicle - Open Government Portal", Open.canada.ca, 2019. [Online]. Available: https://open.canada.ca/data/en/dataset/f6e7e871-79b7-49e1-90a2-e3c913f1951d. [Accessed: 22- Jul- 2019].

[7]"Opinion Lexicon", Kaggle.com, 2019. [Online]. Available: https://www.kaggle.com/nltkdata/opinion-lexicon. [Accessed: 22- Jul- 2019].