Twitter Data Analysis Using NLP Techniques

By,
Ramya Sree V

BackGround

- Tweets provide the feedback to any product / a disease outbreak and therefore inherently useful.
- Every CoronaVirus tweet is summarized by its sentiment score.

Goal

- To Build a classifier that understand the essence of the tweet.

Techniques Used in NLP

Exploration of Natural Language Processing including:

- **Lemmatization & Tokenization**
- **Word Embedding**
- **Topic Modelling**
- **Automatic Summarization**

NLP Preprocessing

In this step we start by Cleaning the tweets, know number of positive, negative & neutral tweets and provide score for each comment.

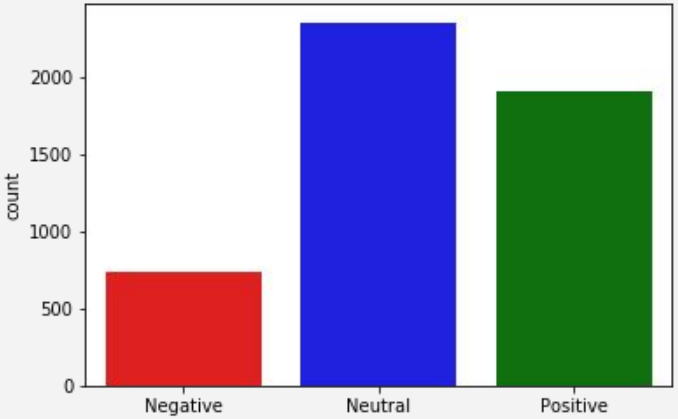
1. Cleaned_tweets = It keeps only spaces, alphabets and numbers and remove the rest .

2. Comments = Each word in the tweet has been split and check the sentiment of each word whether it is positive , negative , & neutral.

3. Score = The sentiment score has been provided for each tweet.

Data Table & Sentiment Analysis Graph

	date_time	tweets	score	Comments
0	2020-03-04 08:43:17	b rt ifccworldlab2020 is postponed due to grow...	0.005682	Positive
1	2020-03-04 08:43:17	b rt is anyone else xe2 x80 x99a town idiots s...	-0.277778	Negative
2	2020-03-04 08:43:16	b rt please retweet ndon t panic n coronavirus...	0.000000	Neutral
3	2020-03-04 08:43:16	b rt chinese scientists have achieved consider...	0.050000	Positive
4	2020-03-04 08:43:16	b reduce your risk from coronavirus	0.000000	Neutral



Count of Sentiments in the Twitter Data

Extracting Root Word

- Lemmatization technique helps to reduce tokens to their root word.
- We used WordNetLemmatizer from the Natural Language Toolkit .
- Lemmatization only applies to each word but it is dependent on sentence structure to understand context.

```
def lemmatize_word (tagged_token):  #returns lemmatized wo
    root = []
    for token in tagged_token:
        tag = token[1][0]
        word = token[0]
        if tag.startswith ('V'):
            root.append (L.lemmatize(word, wordnet.VERB))
        elif tag.startswith ('J'):
            root.append (L.lemmatize(word, wordnet.ADJ))
        elif tag.startswith ('R'):
            root.append (L.lemmatize(word, wordnet.ADV))
        elif tag.startswith ('N'):
            root.append (L.lemmatize(word, wordnet.NOUN))
```

Normalization & Tokenization

Normalizing the Tweets

- **Here each tweet will be normalized from UTF-8 to ASCII encoding. It removes syllables in the tweets and return tweets in simple language.**

Tokenization

- **From cleaned tweets we take out corpora which is simply a collection of tweets. Each tweet is transformed into the list of words.**
- **Removing the 2 letter tokens.**

Count - Based Feature Engineering

In this project we are not using ML, but In order for a ML model the document must first be vectorized. This simply means that the input has to be converted into containers of numerical values.

Bags of Words Model:

The classical approach in expressing text as a set of features is getting the token frequency. Each entry to the dataframe is a document while each column corresponds to every unique token.

```
Word: concern, Frequency: 1
Word: coronavirus, Frequency: 1
Word: date, Frequency: 1
Word: grow, Frequency: 1
Word: ifccworldlab2020, Frequency:
Word: postpone, Frequency: 1
Word: restriction, Frequency: 1
Word: travel, Frequency: 1
Word: worldwide, Frequency: 1
```

TF - IDF Model

(Term Frequency-Inverse Document Frequency)

- The TF-IDF approach assigns continuous values instead of simple integers for the token frequency.
- Words that appear frequently overall tend to not establish saliency in a document, and are thus weighted lower.
- Words that are unique to some documents tend to help distinguish it from the rest and are thus weighted higher.
- The tf-idf weighting is based on our bow variable.

```
Word: concern, Weight: 0.264
Word: coronavirus, Weight: 0.021
Word: date, Weight: 0.348
Word: grow, Weight: 0.360
Word: ifccworldlab2020, Weight: 0.485
Word: postpone, Weight: 0.314
Word: restriction, Weight: 0.422
Word: travel, Weight: 0.262
Word: worldwide, Weight: 0.314
```

Word Embedding

The Word to vector technique Actually embeds meaning in vectors by quantifying how often the word appears with the vicinity of the given set of words.

	0	1	2	3	4	5	6	7	8	9	...	90	91	92
ifccworldlab2020	0.042135	-0.070845	-0.092072	0.000869	-0.020924	-0.090033	-0.019413	0.116442	0.178580	-0.196573	...	0.016999	-0.034277	0.152210
postpone	-0.566481	0.671383	0.310850	0.556309	0.315621	-0.605398	-1.426707	0.615272	1.245851	-1.310820	...	0.353044	0.419243	1.587910
grow	-0.037534	0.053854	-0.190414	0.514681	0.252310	0.002965	-0.123209	0.476202	0.422286	-0.773236	...	0.201793	0.138699	-0.020013
concern	-0.023550	0.065430	0.703482	0.046636	0.244423	1.284435	-0.338893	0.243905	0.400561	-2.400297	...	-0.416512	1.381591	1.694905
coronavirus	-0.327197	-1.186721	-0.214381	0.046531	-0.213391	-0.898435	0.504185	-0.642408	0.394273	-0.984997	...	0.517990	0.263681	0.028457

Topic Modelling - LDA

Latent Dirichlet Allocation is a generative, probabilistic model for a collection of documents, which are represented as mixture of latent topics, where each topic is characterized by a distribution over words.

I have taken the top 10 words that are salient to the first group of coronavirus tweets.

```
coronavirus 0.06854944
have 0.035320252
case 0.029019479
from 0.023930155
india 0.015502137
italian 0.013833896
death 0.0107633965
tourist 0.010283922
break 0.009345112
positive 0.008635188
```

LDA model output

```
coronavirus 0.06854944
have 0.035320252
case 0.029019479
from 0.023930155
india 0.015502137
italian 0.013833896
death 0.0107633965
tourist 0.010283922
break 0.009345112
positive 0.008635188
```

```
Topic 1:
 coronavirus, 0.06854943931102753
 have, 0.03532025218009949
 case, 0.02901947870850563
 from, 0.023930154740810394
 india, 0.015502137131989002

Topic 2:
 coronavirus, 0.050107281655073166
 this, 0.02387525700032711
 hand, 0.022709470242261887
 your, 0.020889196544885635
 with, 0.01761789433658123

Topic 3:
 mask, 0.03198189660906792
 face, 0.02983640879392624
 coronavirus, 0.02566242404282093
 medical, 0.0241343155503273
 include, 0.019403532147407532
```

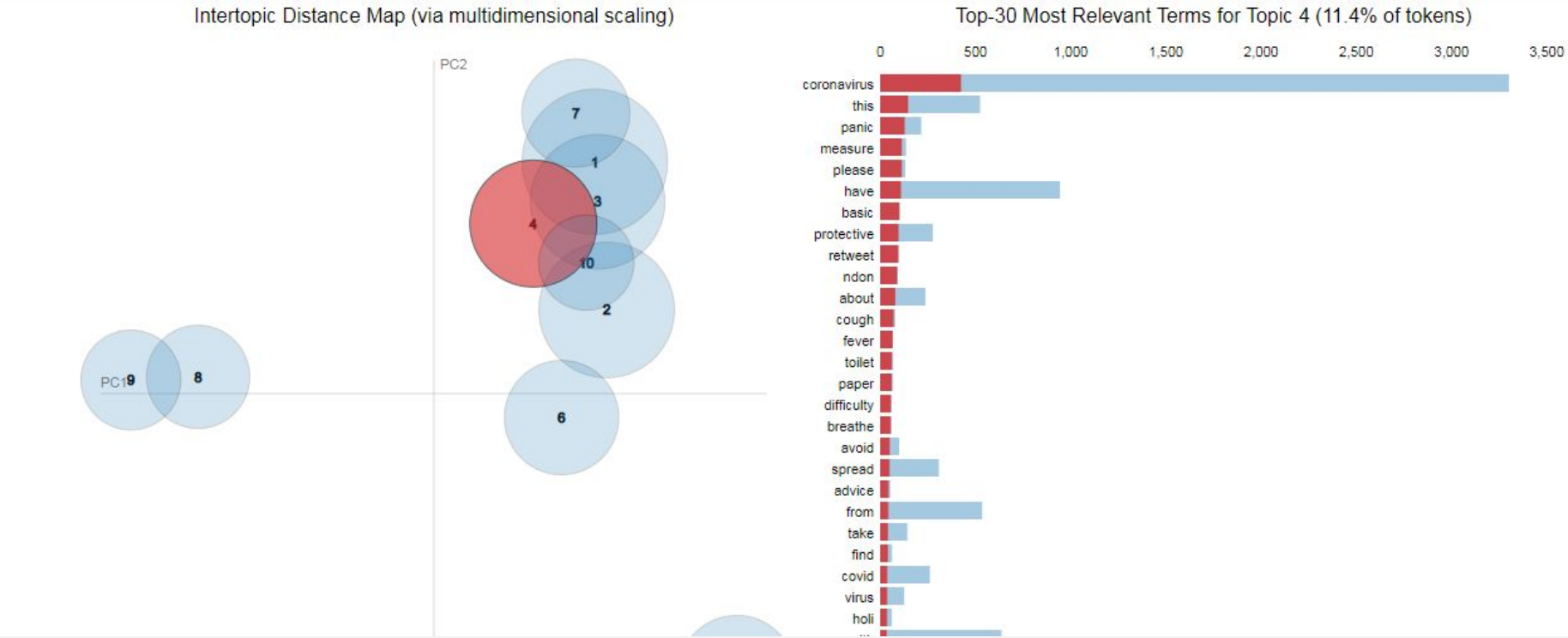
```
Topic 4:
 coronavirus, 0.0735180675983429
 there, 0.04583359137177467
 with, 0.03352723270654678
 shake, 0.028018027544021606
 hand, 0.02687731571495533

Topic 5:
 coronavirus, 0.05519440397620201
 case, 0.03390083834528923
 health, 0.027128534391522408
 confirm, 0.026143180206418037
 have, 0.0186324380338192

Topic 6:
 coronavirus, 0.04758588597178459
 this, 0.02752882055938244
 have, 0.021520916372537613
 korea, 0.01920424774289131
 million, 0.015832284465432167

Topic 7:
 coronavirus, 0.08107677847146988
 case, 0.022768888622522354
 with, 0.01758766360580921
 spread, 0.014338459819555283
 from, 0.012655721977353096
```

Visualization Using PyLDAvis



Automatic Summarization

- **Text Summarization: It is one of the applications in NLP. It summarizes whole article, document or book into a few lines.**
- **Basically it provides insights of the whole document**

Beautiful Soup is a Python library for pulling data out of HTML and XML files. It works with your favorite parser to provide idiomatic ways of navigating, searching, and modifying the parse tree.

Step - by step process

- Summarizing on topic - CoronaVirus
- All the sentences in the paragraph will be collected in the list.
- Tokenization is the process of breaking a stream of text up into words, phrases, symbols, or other meaningful elements called tokens. The list of tokens becomes input for further processing such as parsing or text mining.
- Summary represent most important and relevant information from the original content.

Summarized Text

The genome size of coronaviruses ranges from approximately 27 to 34 kilobases, the largest among known RNA viruses.The name coronavirus is derived from the Latin corona, meaning "crown" or "halo", which refers to the characteristic appearance reminiscent of a crown or a solar corona around the virions (virus particles) when viewed under two-dimensional transmission electron microscopy, due to the surface covering in club-shaped protein spikes.

Summary of the webpage : https://en.wikipedia.org/wiki/Coronavirus

References

Social Medicine: Twitter in Healthcare:
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6025547/

Tweeting About Mental Health - Big Data Text Analytics of twitter for Public Policy:
http://www.rand.org/pubs/rgs_dissertations/RGSD391.html

A Tool For Monitoring and Analyzing Healthcare Tweets:

https://www.semanticscholar.org/paper/A-Tool-for-Monitoring-and-Analyzing-HealthCare-Ali-Magdy/700e490db82e9105c05144137e57415174ac43fd

Business Outcome

- **The data from SNSs can be used effectively to track disease outbreaks & provide necessary warnings .**
- **Twitter share a common goal of changing a perception of medicine from black box to something more accessible & allow patients to understand conditions & make informed decisions.**

Thank you