Author : Sajal Sinha , Arifuddin Atif

Course Title: Corona Virus Tweet Analysis

Keywords: Machine Learning, Sentimental Analysis, Logistic Regression, Tf-IDF, Count Vectorisation

Abstract:

Due to large scale rise in cases of Covid-19, lockdown was imposed in many countries for much longer period which increased average individual time on social media platform. Many people suffered from depression, anxiety and stress due to continuous shutdown. Twitter is a social media platform which is used by most of the people and so by analysing ones tweet we can get a certain estimate on person's sentiment. So we are provided with two months of labelled tweet data and we were expected to prepare a model which can categorise any statements as positive, negative or neutral.

1. Problem Statement

We have data of 41157 tweets of two months I.e of March and April with manual tagging and our main objective to build a classification model that will help in categorising sentiment behind any tweet. In our Data we've following columns:

A) Location :- This Column gives location from which the tweet was made.

B) UserName :- Its a code by which the person is registered in twitter database

C) ScreenName :- A code which represents the on-screen name of user.

D) TweetAt :- Date on which the tweet was made.

E) Original Tweet :- Original Tweet before any prepousseing.

F) Sentiment :- It shows sentiment of the tweet made by person I.e positive, negative etc.

G) Processed_text ;- Text that is used for model training.

2. **Introduction**

The Data has been labelled with 5 different sentiment, Positive, Extremely Positive, Neutral, Negative and Extremely Negative. On the basis of this data we will train our model to understand sentiment of different tweets and then our model can be used for further classification. This analysing of text in-order to know the sentiment behind it is know as Sentiment Analysis.

2.1 **Why Sentimental Analysis?**

Sentiment analysis is extremely useful in social media monitoring as it allows us to gain an overview of the wider public opinion behind certain topics.

The applications of sentiment analysis are broad and powerful. The ability to extract insights from social data is a practice that is being widely adopted by organisations across the world. Shifts in sentiment on social media have been shown to correlate with shifts in the stock market.

The Obama administration used sentiment analysis to gauge public opinion to policy announcements and campaign messages ahead of 2012 presidential election. Being able to quickly see the sentiment behind everything from forum posts to news articles means being better able to strategise and plan for the future.

It can also be an essential part of your market research and customer service approach. Not only can you see what people think of your own products or services, you can see what they think about your competitors too. The overall customer experience of your users can be revealed quickly with sentiment analysis, but it can get far more granular too. The ability to quickly understand consumer attitudes and react accordingly is something that Expedia Canada took advantage of when they noticed that

there was a steady increase in negative feedback to the music used in one of their television adverts.

3. **Steps Involved**

1) Exploratory Data Analysis :- EDA helps us to understand data in much easier way. It gives us some valuable information that is helpful in model building as well as useful in outside-model-work. In our data we plot bar plots, piechart in order to gain insights that might be useful to understand data/people.

2) Null Value Treatment :- It is said that data with **more than 40% null values** should be dropped as they only drag down the model if used. If treated then won't have much effect either and so getting any column at the beginning and to clear it out is very helpful.

3) Preprocessing text :- The text which has been used for model preparation doesn't contain stop words, punctuation marks, hashtags, links, etc in them. The objective of this step is to clean noise those are less relevant to find the sentiment of tweet. If we skip this step then there is high chance that we are working with noisy and inconsistent data.

3.1) Removing Twitter Handles (@user) :- The tweets contain lots of twitter handles (@user), that is how a Twitter user acknowledged on Twitter. We will remove all these twitter handles from the data as they don't convey much information.

3.2) Removing Punctuations, Numbers, and Special Characters :- Punctuations, numbers and special characters do not help much. It is better to remove them from the text just as we removed the twitter handles. Here we will replace everything except characters and hashtags with spaces.

3.3) Removing Short Words

We have to be a little careful here in selecting the length of the words which we want to remove. So, I have decided to remove all the words having length 3 or less. For example, terms like “hmm”, “oh” are of very little use. It is better to get rid of them.

3.4) Tokenization

Tokens are individual terms or words, and tokenization is the process of splitting a string of text into tokens.

3.5) Stemming

Stemming is a rule-based process of stripping the suffixes (“ing”, “ly”, “es”, “s” etc) from a word. For example, For example – “play”, “player”, “played”, “plays” and “playing” are the different variations of the word – “play”.

4) Extracting Features from Cleaned Tweets

To analyze a preprocessed data, it needs to be converted into features. Depending upon the usage, text features can be constructed using assorted techniques – Bag-of-Words, TF-IDF, and Word Embeddings.

4. **Algorithms**

1) Bag-of-Words Features

Bag-of-Words is a method to represent text into numerical features. Consider a corpus (a collection of texts) called C of D documents {d1,d2…..dD} and N unique tokens extracted out of the corpus C. The N tokens (words) will form a list, and the size of the bag-of-words matrix M will be given by D X N. Each row in the matrix M contains the frequency of tokens in document D(i).

Let us understand this using a simple example. Suppose we have only 2 document

D1: He is a lazy boy. She is also lazy.

D2: Smith is a lazy person.

The list created would consist of all the unique tokens in the corpus C.

= [‘He’,’She’,’lazy’,’boy’,’Smith’,’person’]

Here, D=2, N=6

The matrix M of size 2 X 6 will be represented as –

	He	She	lazy	boy	Smith	person
D1	1	1	2	1	0	0
D2	0	0	1	0	1	1

Now the columns in the above matrix can be used as features to build a classification model. Bag-of-Words features can be easily created using sklearn’s CountVectorizer function. We will set the parameter max_features = 1000 to select only top 1000 terms ordered by term frequency across the corpus.

2) TF-IDF Features

This is another method which is based on the frequency method but it is different to the bag-of-words approach in the sense that it takes into account, not just the occurrence of a word in a single document (or tweet) but in the entire corpus.

TF-IDF works by penalizing the common words by assigning them lower weights while giving importance to words which are rare in the entire corpus but appear in good numbers in few documents.

Let’s have a look at the important terms related to TF-IDF:

- TF = (Number of times term t appears in a document)/(Number of terms in the document)
- IDF = log(N/n), where, N is the number of documents and n is the number of documents a term t has appeared in.
- TF-IDF = TF*IDF

Model Building: Sentiment Analysis

We are now done with all the pre-modeling stages required to get the data in the proper form and shape. Now we will be building predictive models on the dataset using the two feature set — Bag-of-Words and TF-IDF.

We will use logistic regression to build the models. It predicts the probability of occurrence of an event by fitting data to a logit function.

The following equation is used in Logistic Regression:

$$\log\left(\frac{p}{1-p}\right) = \beta_o + \beta(\text{Age})$$

5. **Model performance**

1) Confusion Matrix :- The confusion matrix is a table that determines how successful a model is at prediction.

2) Precision/Recall :- Precision is ratio of correct predictions to the overall number of positive predictions: TP/TP+FP.

Recall is the ratio of correct positive predictions to the overall number of positive examples in the set: TP/FN+TP

3) Accuracy :- Accuracy is given by the number of correctly classified examples divided by the total number of classified examples.

4) f1 score :- It considers both Precision and Recall of the test to compute score, F-Score is the Harmonic mean of precision and recall. This will tell you how your system is performing.

6. **Conclusion**

We performed all the step as mentioned and achieved an accuracy of

7. **References**

a) https://www.brandwatch.com/blog/understanding-sentiment-analysis/

b) https://www.securekloud.com/blog/benchmarking-sentiment-analysis-systems/

c) https://www.analyticsvidhya.com/blog/2018/07/hands-on-sentiment-analysis-dataset-python/