Tweet Sentiment Extraction

Problem Statement

***"My ridiculous dog is amazing."* [sentiment: positive]**

With all of the tweets circulating every second it is hard to tell whether the sentiment behind a specific tweet will impact a company, or a person's, brand for being viral (positive), or devastate profit because it strikes a negative tone. Capturing sentiment in the language is important in these times where decisions and reactions are created and updated in seconds. But, which words actually lead to the sentiment description? In this project we will try to pick out the part of the tweet (word or phrase) that reflects the sentiment.

Solution Approach

As for a problem, there are various methods to solve it. Same in this project there are various models to approach this competition, but we decided to work with NER(Name Entity Recognition) model. The specific reason for selecting this model is only that we all are new in the field of NLP and this model is good for a start.

Named Entity Recognition (NER)

Named Entity Recognition (NER) is an application of Natural Language Processing (NLP) that processes and understands large amounts of unstructured human language. Also known as entity identification, entity chunking and entity extraction. NER extraction is the first step in answering questions, retrieving information and topic modeling. NER seeks to locate and classify named entity mentioned in unstructured text into pre-defined categories such as person names, organizations, locations, medical codes, time expressions, quantities, monetary values, percentages, etc. So in our case, the predefined categories would be positive, negative and neutral. We are going to obtain pre-trained models for each of the trained sentiments and apply ner to each of them. It has been observed in the other kernels that having a single model is not quite effective. So we have used separate models to each of the sentiment labels. The trained models that we have used are also available in the repo.

How to build NER model

This model is built with the help of NLTK and spaCY packages.

Why NLTK is used for building NER model?

NLTK (Natural Language Toolkit) is a wonderful Python package that provides a set of natural language corpora and APIs to an impressive diversity of NLP algorithms. It's easy to use, complete, and well documented. Of course, it's free, open-source, and community-driven.

Why spaCY is used for building NER model?

It can be used to build information extraction or natural language understanding systems or to pre-process text for deep learning. It provides a default model that can recognize a wide range of named or numerical entities, which include person, organization, language, event, etc.

After going through this quick tutorial about spaCY we decided to build our model by using spaCY. This website gave us an understanding, how to train our model using spaCY. This guide also helped us with the code.

In our model, we had used text as a selected_test for all our neutral sentiments due to their high Jaccard similarity. We had trained two different models for positive and negative tweets. We had not preprocessed the data because selected_text contains the raw text.

With this model, we had achieved an accuracy of 66.4.

Notebook from Kaggle

Twitter sentiment Extraction-Analysis, EDA and Model -
https://www.kaggle.com/tanulsingh077/twitter-sentiment-extaction-analysis-eda-and-model

NER - training using spacy (Ensemble) -

https://www.kaggle.com/rohitsingh9990/ner-training-using-spacy-ensemble

Article's
https://towardsdatascience.com/building-a-question-answering-system-part-1-9388aadff507

https://medium.com/inside-machine-learning/what-is-a-transformer-d07dd1fbec04

https://medium.com/@manivannan_data/how-to-train-ner-with-custom-training-data-using-spacy-188e0e508c6

https://medium.com/@b.terryjack/nlp-pretrained-named-entity-recognition-7caa5cd28d7b

https://towardsdatascience.com/named-entity-recognition-with-nltk-and-spacy-8c4a7d88e7da

Additional Sources

We also went through some of the lecture videos, notes, and slides of Cs224n. A course on Coursera helped us to write an NLP code using TensorFlow. We had also learned a bit about how to write NLP code using PyTorch, here is a link to the tutorial which we went through. We had also read articles about RNN model, Bidirectional RNN, GRU, LSTM, etc.

Future Prospects

For building our next model we will switch to the Transformers. Because all other participants who had used transformers like BERT, ALBERT, RoBERTA had achieved more accurate models and are at the top of the leaderboard. So, for building a model which will be more accurate than the previous model we had to switch to Transformers.

TRANSFORMER

Transformers are a type of neural network architecture that has been gaining popularity. Transformers were developed to solve the problem of **sequence transduction,** or **neural machine translation.** That means any task that transforms an input sequence to an output sequence. This includes speech recognition, text-to-speech transformation, etc.

We have decided to go through the research papers of a different transformer like BERT, ALBERT, RoBERTA etc.

.