Applied Machine Learning and Data Science Internship 2020, IIT Kanpur.

Capstone Project Interim Report

Tweets Sentiment Analysis

Jatin Mishra, 3rd July 2020.

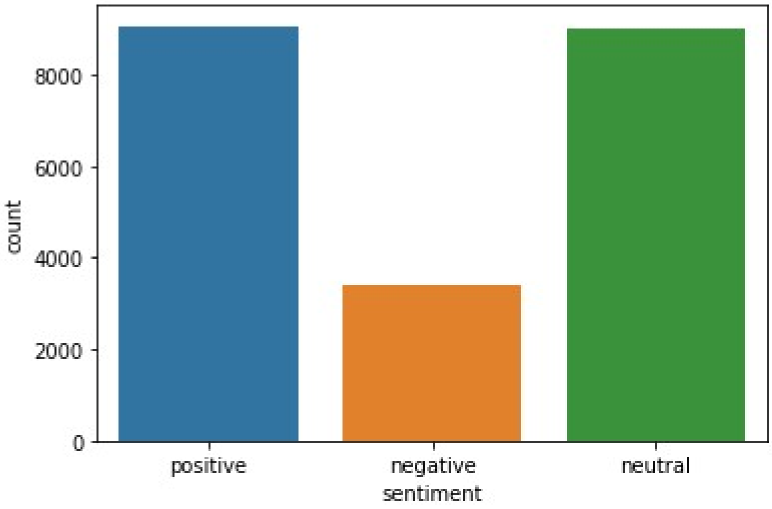
Introduction

As the name implies this problem is about building a model that can classify tweets into three classes namely “Positive”, “Negative” and “Neutral”. This is a text classification problem which can be solved using Natural Language Processing.
Sentiment Analysis is one of the most important problems approached in Natural Language Processing as it has numerous real-world use cases like analyzing movie reviews, product reviews, customer feedback, etc list goes on. It can be used pretty much anywhere where the requirement is to get opinions by analyzing textual data. The textual data can be things like tweets (as our problem here), blogs, reviews, forums, social media, news, etc.

Difficulties

Here we will discuss briefly some difficulties in creating a model to analyze tweets to classify them based on sentiments.

1. **Data imbalance.**
   The first difficulty we encounter in the provided training data is class imbalance.



We can see the negative class samples are less than half of the other classes. This can lead to a model leaning towards predicting the bigger classes (“positive”, “neutral”) and ignoring the smaller ones (“negative”). We can reduce this effect on our model by using **oversampling** (repeating some samples of minority class to match up with the bigger classes) or **downsampling** (reducing samples of majority classes to match up with minority class). One thing to note here is both these approaches should be applied before cross-validation split to avoid overfitting.

2. **Data Cleaning**
   Probably the most important difficulty is data cleaning or data preprocessing. It is simply a process to remove unwanted data from the text samples before we vectorize them to feed them to a neural network. The better the data cleaning the easier is for the network to recognize necessary features that contribute the most in our prediction.

Feasible Approaches

Here we briefly pin down the feasible approaches to create a sentiment analysis model.

1. **Logistic Regression**
   We can use Bag Of Word or TF-IDF Vectorizer with Logistic Regression and use a Softmax function to calculate probabilities of each class where the class with the highest probability is the predicted. This approach is one of the most rudimentary approaches used for text classification which provides somewhat good results.
2. **Recurrent Neural Networks (RNN)**
   RNN specifically LSTM (Long Short Term Memory) is a good choice in sentiment analysis because “sequence is very important in textual data”. The previous approach does not take into account the sequence of words in the sample which limits its performance. LSTM performs better here because they preserve the sequence of words in the text to predict sentiments.
3. **Combination of CNN and LSTMs**
   The combination of CNN and LSTMs performs very well because CNNs are very good at identifying local features. In case of textual data phrases like “very bad”, “very impressive”, “disappointing”, etc. can clearly define the sentiment of a statement. Also, CNN is much faster than RNNs. Some cases may require to consider the longer sequence of data to capture better meaning but it is essentially not required every time. Here local feature extraction like phrases “very bad”, “great”, etc can provide better sentiment prediction at a much lesser computation.
   Combining the speed, good local feature extraction of CNNs and context dependencies, sequence prediction specialty of RNNs result in a very good model (sometimes State Of the Art performances) of NLP tasks such as text classification (here sentiment analysis specifically).