TWEET SENTIMENT EXTRACTION

NIKITHA KRISHNA VEMULAPALLI[1], NAUREEN FIRDOUS[1], VIJAYANTIKA INKULLA[1]

[1] UNIVERSITY OF SOUTHERN CALIFORNIA

ABSTRACT

With a huge number of tweets being circulated every second, it has become vital for people and organizations to analyze if they will impact their profit. Apart from capturing the sentiment of those tweets, it is equally important to understand the part of tweet that contributes to that sentiment. This will help the organizations to identify if a particular product is causing their profit to decrease.

Extracting the sentiment can be done by training on the given data, using an ensemble learning model that utilizes random forest classifier, Naive Bayes classifier, Support Vector Machine and Multi-layer Perceptron. The features for this training data can be obtained using Term Frequency- Inverse Document Frequency. The phrase for the sentiment can be obtained by using K-Means to identify clusters in each sentiment label (positive, negative, neutral). Thus, our goal is to identify the sentiment of a tweet and extract its corresponding phrase.

INTRODUCTION

Sentiment analysis is the process of extracting emotions or opinions from a piece of text for a given topic. It allows us to understand the attitudes, opinions and emotions in the text. In it user's likes and dislikes are captured from web content. It involves predicting or analyzing the hidden information present in the text. This hidden information is very useful to get insights of user's likes and dislikes. The aim of sentiment analysis is to determine the attitudes of a writer or a speaker for a given topic. Sentiment analysis can also be applied to audio, images and videos. Today internet has become the major part of our life. Most of the people use online blogging sites or social networking sites to express their opinions on certain things.

Based on the sentiment conveyed in the text, subjective text can be classified into positive, negative and neutral. Positive is a kind of sentiment which denotes positivity like 'good', 'happy', 'great'. Negative is a kind of sentiment denoted negativity like 'bad', 'worst', 'not good'. Neutral is a kind of sentiment denotes equal amounts of positive and negative polarity or does not convey any positive or negative feeling.

Two approaches are mainly used for Sentiment Analysis: Subjective lexicon and Machine Learning approach. Subjective lexicons are collection of words where each word has a score indicating the positive, negative, neutral and objective nature of text. In this approach, for a given piece of text, aggregation of scores of subjective words is performed i.e. positive, negative, neutral and objective word scores are summed up separately. In the end there are four scores. Highest score gives the overall polarity of the text. Machine learning is an automatic classification technique. Classification is performed using text features. Features are extracted from text. It is of two types- supervised and unsupervised. When the model is trained on data with predefined labels, it is supervised learning. On the other hand, if the model is trained on data with no predefined labels, it is called unsupervised learning. It groups the data based on a similarity measure.

For this application, the sentiment analysis is performed using supervised learning. The training data for developing the model contains the following fields:

TRAINING DATA

Text_ID
Text
Selected_text
Sentiment

The "Text" field refers to the Tweet text. The "Sentiment" field refers to sentiment of the text which can be positive, negative or neutral. The "Selected_text" field contain a part of the data from the "Text" field that corresponds to the given sentiment.

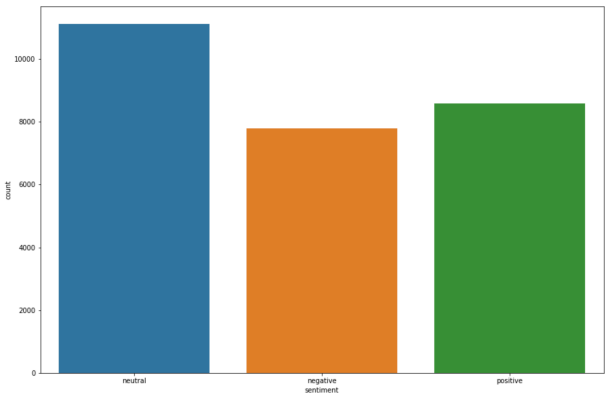
The training data is used to develop a machine-learning model for analyzing the sentiment. An ensemble classifier is used which comprises of Naive Bayes, Random Forest classifier, Support Vector Machine, and Multilayer Perceptron model. This model is used to predict the sentiment of text in the test data.

After predicting the sentiment of the data, the "selected_text" is predicted using K-Means algorithm.

PREPROCESSING AND ANALYSIS OF TRAINING DATA

The training data contains some null entries. As part of preprocessing, the null entries are removed and all the fields are converted to a string data type.

On analyzing the training data, it is observed that the number of records with "neutral" sentiment is higher compared to "positive" and "negative".



PREDICTION OF SENTIMENT

The sentiment of the text in test data is predicted by developing a Machine Learning model using the training data. For learning a model, following preprocessing steps are performed on the "text" field data in the training data and test data:

- Tokenization is performed on the text and the words that do not begin with letters are removed.
- Lemmatization is performed on the text. Lemmatization is the process of converting a word to its base form. The difference between stemming and lemmatization is, lemmatization considers the context and converts the word to its meaningful base form, whereas stemming just removes the last few characters, often leading to incorrect meanings and spelling errors.
- The text from both training and test data is used to find their Term Frequency-Inverse Document Frequency. The text from training data and test data are then converted to their respective Term Frequency- Inverse Document Frequency vector.

After the preprocessing, the Term Frequency-Inverse Document Frequency vector of the training data is used as the input for an ensemble learning model. The ensemble learning model uses Naive Bayes classifier, Random Forest classifier, Support vector machine with both a radial basis function kernel and a polynomial kernel, and a Multi-layer perceptron learning algorithm. The training data is trained on the individual classifiers as well as the ensemble model. The accuracies obtained are:

Accuracy: 0.63 (+/- 0.01) [Random Forest]

Accuracy: 0.63 (+/- 0.01) [SVC-RBF]

Accuracy: 0.53 (+/- 0.00) [SVC-Poly]

Accuracy: 0.58 (+/- 0.00) [Naive Bayes]

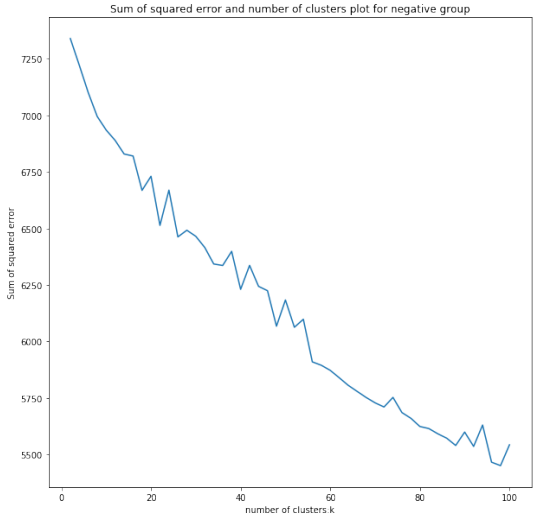
Accuracy: 0.61 (+/- 0.01) [Ensemble]

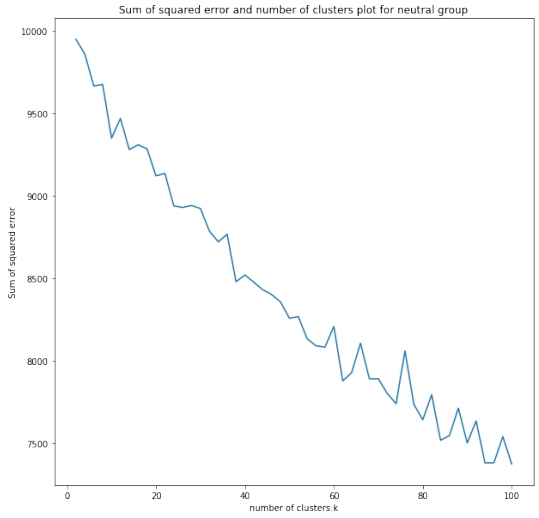
The model with the best accuracy(Random-Forest) is used to predict the sentiment of test data.

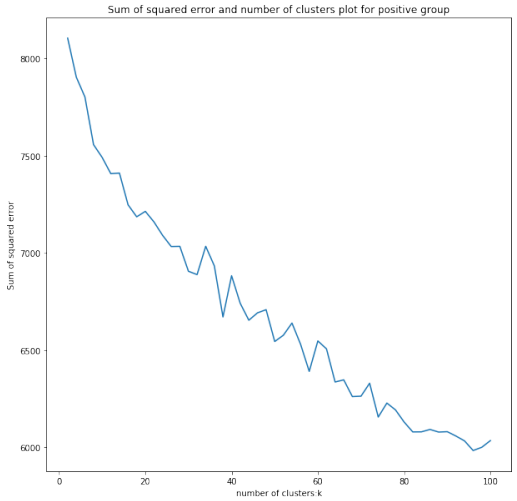
SENTIMENT TEXT EXTRACTION

The "selected_text" is to be predicted based on the already predicted sentiment. The text from both training and test data is used to find their Term Frequency- Inverse Document Frequency. The text from training data and test data are then converted to their respective Term Frequency- Inverse Document Frequency vector.

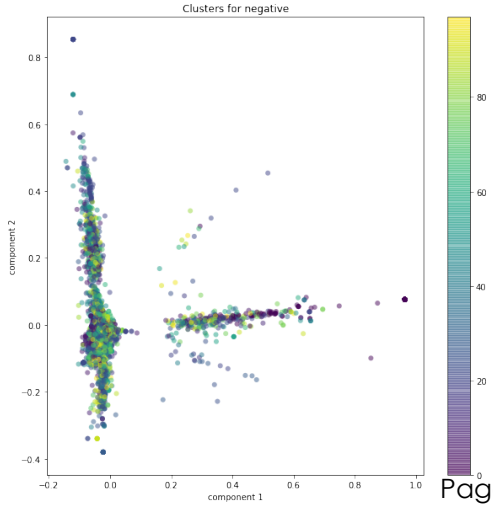
Since we know the sentiment for the text of both training and test data, the records can be grouped based on sentiment. We can predict selected text by identifying similarities in these grouped records separately, So, K-means is a good algorithm to identify clusters in these groups (groups are positive, negative and neutral). For this application, MiniBatchKMeans is used instead of general KMeans, since MiniBatchKmeans speeds up the process. In Mini-batch k-means the most computationally costly step is conducted on only a random sample of observations as opposed to all observations. To perform the k-means, the number of clusters for each group is to be identified. This can be done by running the algorithm against different number of clusters and plotting the sum of squared error against the number of clusters. The optimal number of clusters would be at a position where the sum of squared error is least. The plots obtained are:

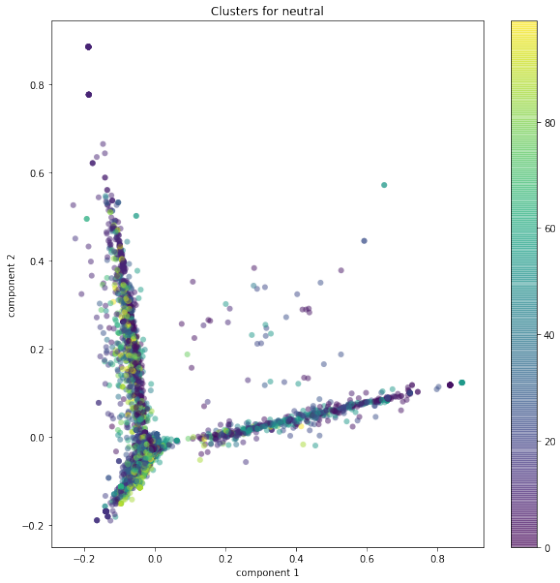


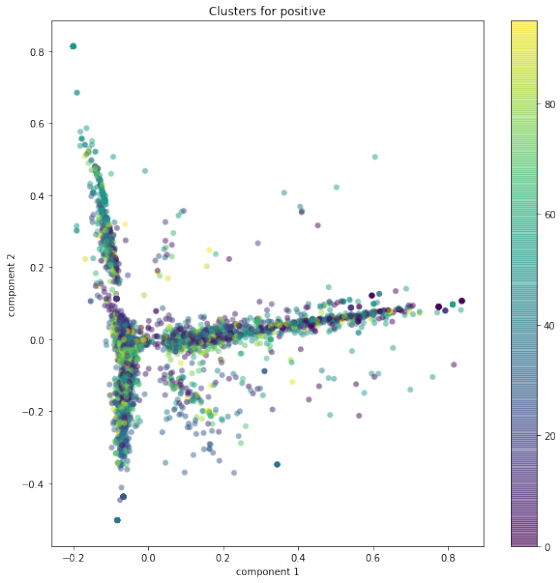




The Mini Batch K-Means is performed using the number of clusters obtained. The clusters that are obtained can be plotted by performing PCA to reduce the number of dimensions to two and plotting them.







The metric that is used to obtain selected_text is jaccard score. The jaccard score is calculated on the "text" and "selected_text" fields for every record of training data.

The "selected_text" for the test data is predicted by identifying the cluster it belongs to by prediction on Mini-Batch K-Means model. This cluster is used to get the top keywords from the cluster and use the words that are in the "text" of that record.

CONCLUSION

Analyzing the tweets is a huge application for businesses. The tweets can be preprocessed by performing tokenization and lemmatization on them and converting into a Term Frequency-Inverse Document Frequency matrix. The sentiment can then be extracted by using an ensemble model for higher accuracy. Random Forest, Support Vector Machine, Naive Bayes algorithms can be used as part of the model. The text corresponding to the sentiment of the tweets can be extracted by building a K-Means model for each of the positive, negative and neutral groups. This analysis of tweets offers a great value to businesses.

REFERENCES

Harpreet Kaur, Veenu Mangat, Nidhi, "A Survey of Sentiment Analysis Techniques", International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC 2017), 2017.

Dr. Pushpak Bhattacharya, "Sentiment Analysis", 1st International Conference on Emerging Trends and Applications in Computer Science, 2013.

Training and testing data, Retrieved from https://www.kaggle.com/c/tweet-sentiment-extraction/data

Sentiment analysis process, https://towardsdatascience.com/creating-the-twitter-sentiment-analysis-program-in-python-with-naive-bayes-classification-672e5589a7ed

CONTRIBUTIONS

1. Nikitha Krishna Vemulapalli: "ensemble.ipynb" and 1/3rd report
2. Naureen Firdous: "Analysis.ipynb" and 1/3rd "model.ipynb", 1/3rd report
3. Vijayantika Inkulla: 2/3rd "model.ipynb" and 1/3rd report