Supervised text classification using Naïve Bayes and Neural networks

Step1: Separate the rows where data[“tweet_intent”]==”NaN”. This is considered as test data

Step2: Text Cleaning using NLTK :

i) removing mentioning that starts with “@”

ii) removing links

iii) remove the # symbol,numbers and convert all words into lower case.

iv) Apply lematization and stemming

v) Remove stop_words

vi) Remove words which occur less than 2 times.

Step3: Draw a Word Cloud to better analyse which words are repeating frequently.

Step4: Now the important step i.e; Convert text data into numerical data.

Method1: sklearn.feature_extraction.text import CountVectorizer

This is also called Bag-of-Words.

Method2: sklearn.feature_extraction.text import TfidfVectorizer

Method3: keras.preprocessing.text import Tokenizer

Step5: Use sklearn.model_selection train_test_split and divide the trian data into X_train and X_test.

Step6: Using Models to analyze the data:

Model1: Naïve Bayes Method: MultinomialNB, BernoulliNB

Observation : Model showed significant deference in accuracy when used CountVectorizer data but no difference when used Tfidf Vectorizer.

Accuracy was around 88%

Model2: Using keras.layers Embedding and LSTM.

Accuracy around 95%

Model3: Using keras.layers SimpleRNN (Recurrent Neural Network)

Accuracy around 95%