Capstone Project Report

Name: Sumit Patel

Course: AI and ML (Batch – AUG 2020)

Duration: 10 months

Text Classification

Problem Statement:

Using vector semantics, we can easily convert a given text into its corresponding vector form. Given any text, first pre process the text and convert it into a vector using BoW methods. Given this vector, implement your own classifier to classify the vector is pre-defined categories. You may use of these datasets for training and for defining the categories:

14 Best Text classification Datasets for Machine Learning

Prerequisites

What things you need to install the software and how to install them:

Python 3.6 This setup requires that your machine has latest version of python. The following url https://www.python.org/downloads/ can be referred to download python. Once you have python downloaded and installed, you will need to setup PATH variables (if you want to run python program directly, detail instructions are below in how to run software section). To do that check this: https://www.pythoncentral.io/add-python-to-path-python-is-not-recognized-as-an-internal-or-externalcommand/. Setting up PATH variable is optional as you can also run program without it and more instruction are given below on this topic. Second and easier option is to download anaconda and use its anaconda prompt to run the commands.

To install anaconda check this url https://www.anaconda.com/download/ You will also need to download and install below 3 packages after you install either python or anaconda from the steps above Sklearn (scikit-learn) numpy scipy if you have chosen to install python 3.6 then run below commands in command prompt/terminal to install these packages:

pip install numpy

pip install pandas

pip install sklearn

pip install tensorflow

pip install nltk

If you have chosen to install anaconda then run below commands in anaconda prompt to install these packages:

conda install -c anaconda numpy

conda install -c anaconda pandas

conda install -c anaconda sklearn

conda install -c anaconda tensorflow

conda install -c anaconda nltk

Importing the libraries and loading dataset.

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import matplotlib.pyplot as plt
import string
import nltk
from nltk.stem import PorterStemmer
from nltk.corpus import stopwords
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, classification_report
from sklearn.svm import SVC
from sklearn.naive_bayes import MultinomialNB
from tensorflow.keras.utils import to_categorical
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers
```

Loading and reading data

```
train = pd.read_csv('../24. Text Classification/Corona_NLP_train.csv')
test = pd.read_csv('../24. Text Classification/Corona_NLP_test.csv')
train.head()
```

	UserName	ScreenName	Location	TweetAt	OriginalTweet	Sentiment
0	3799	48751	London	16-03-2020	@MeNyrbie @Phil_Gahan @Chrisitv https://t.co/i...	Neutral
1	3800	48752	UK	16-03-2020	advice Talk to your neighbours family to excha...	Positive
2	3801	48753	Vagabonds	16-03-2020	Coronavirus Australia: Woolworths to give elde...	Positive
3	3802	48754	NaN	16-03-2020	My food stock is not the only one which is emp...	Positive
4	3803	48755	NaN	16-03-2020	Me, ready to go at supermarket during the #COV...	Extremely Negative

Data pre-processing

```
punctuation = ["'","@","#",",",".","/",":",";",'"',"(",")","_"]
def remove_punctuation(text):
    punctuationfree="".join([i for i in text if i not in punctuation])
    return punctuationfree
```

```
#storing the puntuation free text
train['clean_msg']= train['OriginalTweet'].apply(lambda x:remove_punctuation(x))
train.head()
```

	UserName	ScreenName	Location	TweetAt	OriginalTweet	Sentiment	clean_msg
0	3799	48751	London	16-03-2020	@MeNyrbie @Phil_Gahan @Chrisitv https://t.co/i...	Neutral	MeNyrbie PhilGahan Chrisitv httpstcoiFz9FAn2Pa...
1	3800	48752	UK	16-03-2020	advice Talk to your neighbours family to excha...	Positive	advice Talk to your neighbours family to excha...
2	3801	48753	Vagabonds	16-03-2020	Coronavirus Australia: Woolworths to give elde...	Positive	Coronavirus Australia Woolworths to give elder...
3	3802	48754	NaN	16-03-2020	My food stock is not the only one which is emp...	Positive	My food stock is not the only one which is emp...
4	3803	48755	NaN	16-03-2020	Me, ready to go at supermarket during the #COV...	Extremely Negative	Me ready to go at supermarket during the COVID...

```
train['msg_lower']= train['clean_msg'].apply(lambda x: x.lower())
test['msg_lower']= test['clean_msg'].apply(lambda x: x.lower())
train.head()
```

	UserName	ScreenName	Location	TweetAt	OriginalTweet	Sentiment	clean_msg	msg_lower
0	3799	48751	London	16-03-2020	@MeNyrbie @Phil_Gahan @Chrisitv https://t.co/i...	Neutral	MeNyrbie PhilGahan Chrisitv httpstcoiFz9FAn2Pa...	menyrbie philgahan chrisitv httpstcoifz9fan2pa...
1	3800	48752	UK	16-03-2020	advice Talk to your neighbours family to excha...	Positive	advice Talk to your neighbours family to excha...	advice talk to your neighbours family to excha...
2	3801	48753	Vagabonds	16-03-2020	Coronavirus Australia: Woolworths to give elde...	Positive	Coronavirus Australia Woolworths to give elder...	coronavirus australia woolworths to give elder...
3	3802	48754	NaN	16-03-2020	My food stock is not the only one which is emp...	Positive	My food stock is not the only one which is emp...	my food stock is not the only one which is emp...
4	3803	48755	NaN	16-03-2020	Me, ready to go at supermarket during the #COV...	Extremely Negative	Me ready to go at supermarket during the COVID...	me ready to go at supermarket during the covid...

```
unwanted_cols = ['UserName', 'ScreenName', 'Location', 'TweetAt', 'OriginalTweet', 'clean_msg','Sentiment']
```

```
train_X = train.drop(unwanted_cols, axis=1)
train_Y = train['Sentiment']
test_X = test.drop(unwanted_cols, axis=1)
test_Y = test['Sentiment']
```

```
ps = PorterStemmer()
```

```
def pre_process(text):
    text = [word for word in text.split() if word.lower() not in stop_words]
    words = ""
    for i in text:
            words += (ps.stem(i))+" "
    return words
```

```
textFeatures_train = train_X['msg_lower'].copy()
textFeatures_train = textFeatures_train.apply(pre_process)
textFeatures_train[:5]
```

```
0    menyrbi philgahan chrisitv httpstcoifz9fan2pa ...
1    advic talk neighbour famili exchang phone numb...
2    coronaviru australia woolworth give elderli di...
3    food stock one empti pleas dont panic enough f...
4    readi go supermarket covid19 outbreak im paran...
Name: msg_lower, dtype: object
```

```
textFeatures_combined = pd.concat([textFeatures_train, textFeatures_test], axis = 0)
len(textFeatures_combined)
```

```
44955
```

```
vectorizer = TfidfVectorizer(ngram_range=(1, 2))
features_combined = vectorizer.fit_transform(textFeatures_combined)
features_combined.shape
```

```
(44955, 546515)
```

```
train_X = features_combined[:len(textFeatures_train)]
test_X = features_combined[len(textFeatures_train):]
train_X.shape, test_X.shape
```

```
((41157, 546515), (3798, 546515))
```

Training and evaluating the model

```
# Prediction using Support Vector Machine
svc = SVC(kernel='sigmoid', gamma=1.0)
svc.fit(train_X, train_Y)
prediction = svc.predict(test_X)
# accuracy_score(labels_test,prediction)
print(classification_report(test_Y, prediction, target_names = target_names))
```

```
                    precision    recall  f1-score   support

           Neutral       0.72      0.39      0.50       592
          Positive       0.77      0.47      0.58       599
Extremely Negative       0.50      0.56      0.53      1041
          Negative       0.61      0.63      0.62       619
Extremely Positive       0.48      0.67      0.56       947

          accuracy                           0.56      3798
         macro avg       0.62      0.54      0.56      3798
      weighted avg       0.59      0.56      0.56      3798
```