DataGlacier NLP Interns Group

MEMBER DETAILS

- Member 1:
  - Name – Gabriel Aluede
  - Email – ghabyaluede@gmail.com
  - Country – United Kingdom
  - College – Teesside University
  - Specialization – NLP
- Member 2:
  - Name – Yosha Udaya Shriyan
  - Email – yosha.shriyan@gmail.com
  - Country – United Kingdom
  - College – Royal Holloway University of London
  - Specialization – NLP

PROBLEM DESCRIPTION

The term hate speech is understood as any type of verbal, written or behavioural communication that attacks or uses derogatory or discriminatory language against a person or group based on what they are, in other words, based on their religion, ethnicity, nationality, race, colour, ancestry, sex or another identity factor. In this problem, we will take you through a hate speech detection model with Machine Learning and Python.

IDENTIFYING PROBLEMS IN DATA

Pre-process data using the library 'preprocessor'

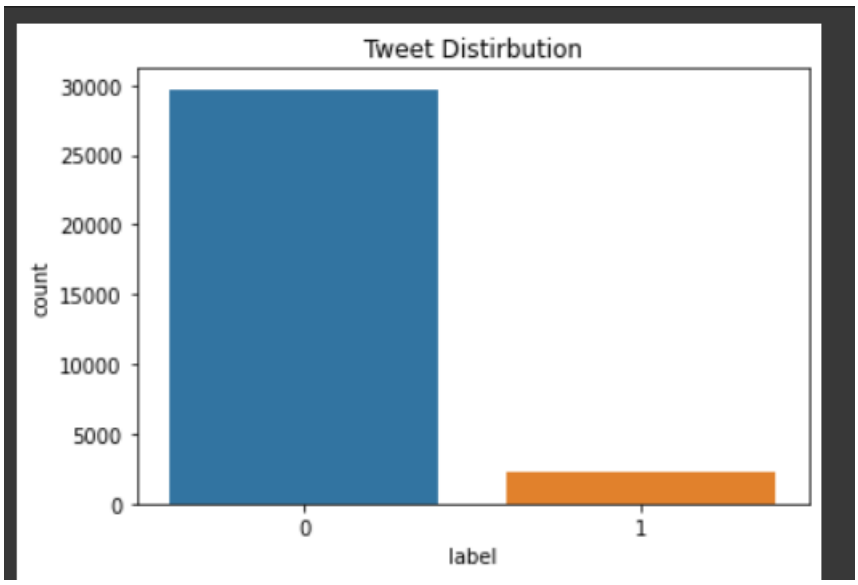
```
import preprocessor as p
REPLACE_NO_SPACE = re.compile("[^a-zA-Z]http\S+\s*RT|cc#\S+@user\S+!\s+")
REPLACE_WITH_SPACE = re.compile("(<br\s/><br\s/?)|(-)|(/)|(:).")
```

Defining function to preprocess each tweet and clean the data

```
def processed_tweets(train_data):
  subArr = []
  for line in train_data:
    #remove punctuation
    tweet = p.clean(line)
    tweet = REPLACE_NO_SPACE.sub("", tweet.lower())
    tweet = REPLACE_WITH_SPACE.sub(" ", tweet)
    subArr.append(tweet)
  return subArr
```

Problem

After cleaning, we visualise the data



We observe that our training data is imbalanced. We will up sample '0' in training set. This will remove every unbiased distribution in the training.

Solution

We start by separating the classes

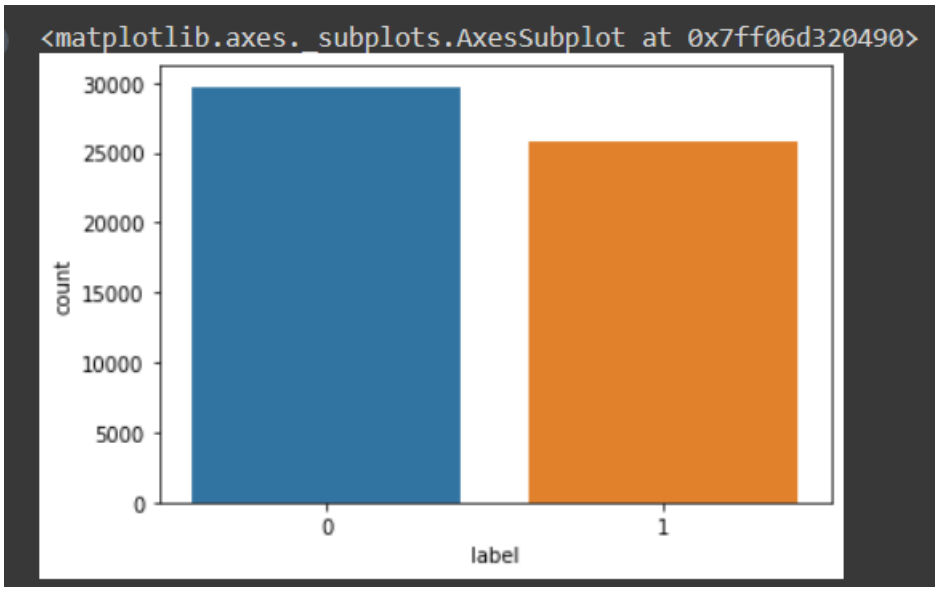
```
#Seperating the the classes
train_0 = train_data[train_data['label']==0]
train_1 = train_data[train_data['label']==1]

#Shape of the separated classes
train_0.shape, train_1.shape
```

```
((29720, 3), (2242, 3))
```

We then create a function to resample the data

```
#Creating a function to resample
upsampled_train_1 = resample(train_1,
                             replace=True,
                             n_samples=25760,
                             random_state=185
                            )
upsampled_train_1.shape
```

Finally, we merge the resampled data with the other majority up-sampled class. We get a more balanced data as the result.



GITHUB REPO LINK

https://github.com/YoshaRHUL/DataGlacierProject