Task Report

Tweet classification task where the tweets could be either Politics or Sports :

realized by :

Boujbair Oussamae

Institut Nationale des Postes et Télécommunication

04 November 2022
Rabat, Morocco

Contents

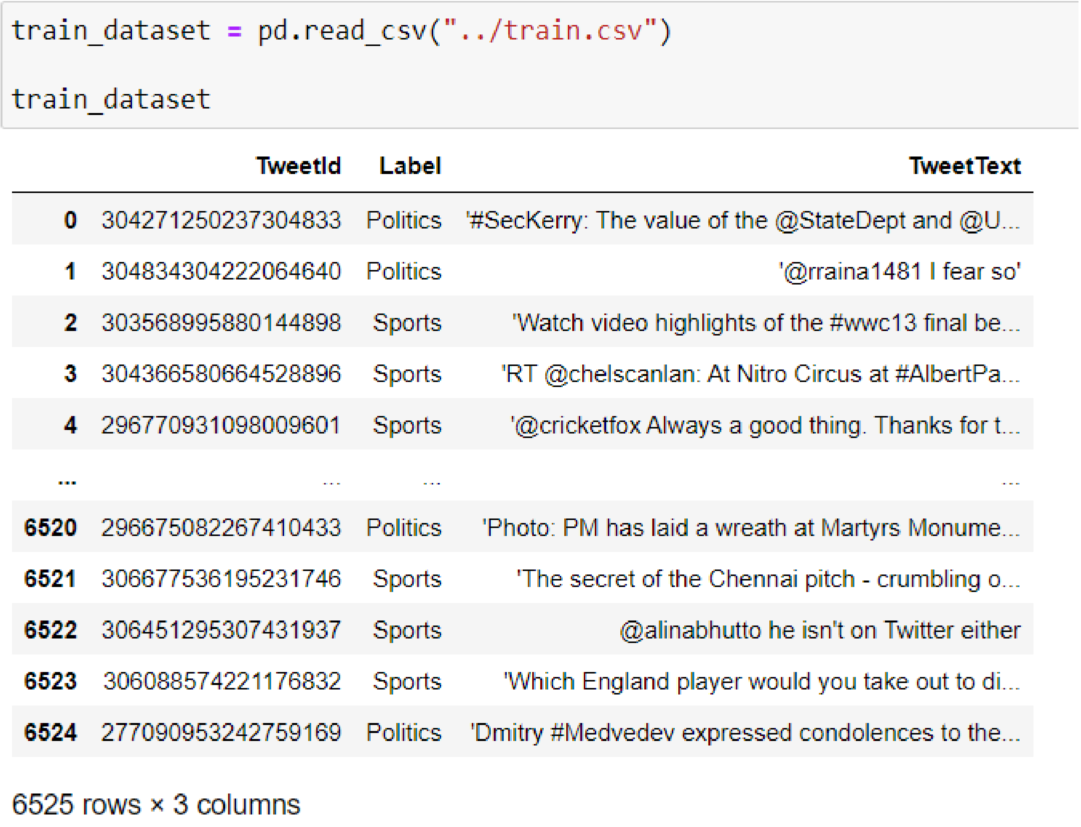
1 Introduction and Background

Twitter is a social networking and micro blogging service on which users post and interact with each other through messages known as “tweets”. It’s ranked as the 6th most popular social networking site and app by Dream Grow as of April, 2020 with an average of 330 million active monthly users.

During a mass casualty event, social networks such as Twitter or Facebook act as a conduit of information. These information include location and type of personal injury, infrastructure damage, donations, advice, and emotional support. Useful information can be harnessed by first responders and agencies to assess damage and coordinate rescue operations. However, the speed and the mass at which the information come in presents a challenge to rescue personnel to discern the relevant ones from extraneous ones. In this light, we want to create a machine learning model that can automatically classify tweets into two categories (sports, politics).

2 Dataset

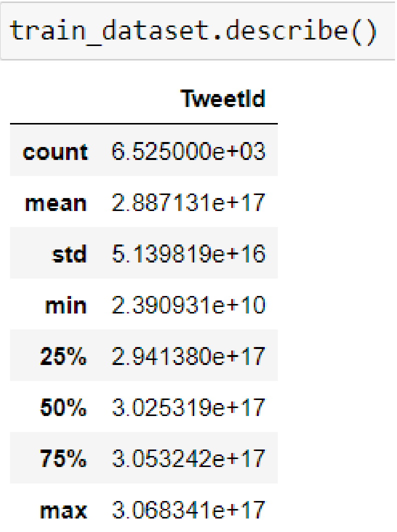
The Train dataset is composed of three columns ”TweetId”, ”Label”, ”TweetText” and 6525 rows.



```
train_dataset = pd.read_csv("../train.csv")

train_dataset
```

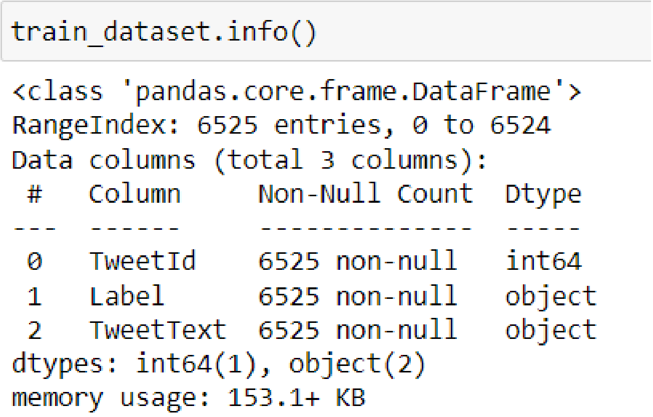
	TweetId	Label	TweetText
0	304271250237304833	Politics	'#SecKerry: The value of the @StateDept and @U...
1	304834304222064640	Politics	'@rraina1481 I fear so'
2	303568995880144898	Sports	'Watch video highlights of the #wwc13 final be...
3	304366580664528896	Sports	'RT @chelscanlan: At Nitro Circus at #AlbertPa...
4	296770931098009601	Sports	'@cricketfox Always a good thing. Thanks for t...
...	...	...	...
6520	296675082267410433	Politics	'Photo: PM has laid a wreath at Martyrs Monume...
6521	306677536195231746	Sports	'The secret of the Chennai pitch - crumbling o...
6522	306451295307431937	Sports	@alinabhutto he isn't on Twitter either
6523	306088574221176832	Sports	'Which England player would you take out to di...
6524	277090953242759169	Politics	'Dmitry #Medvedev expressed condolences to the...

6525 rows × 3 columns



```
train_dataset.describe()
```

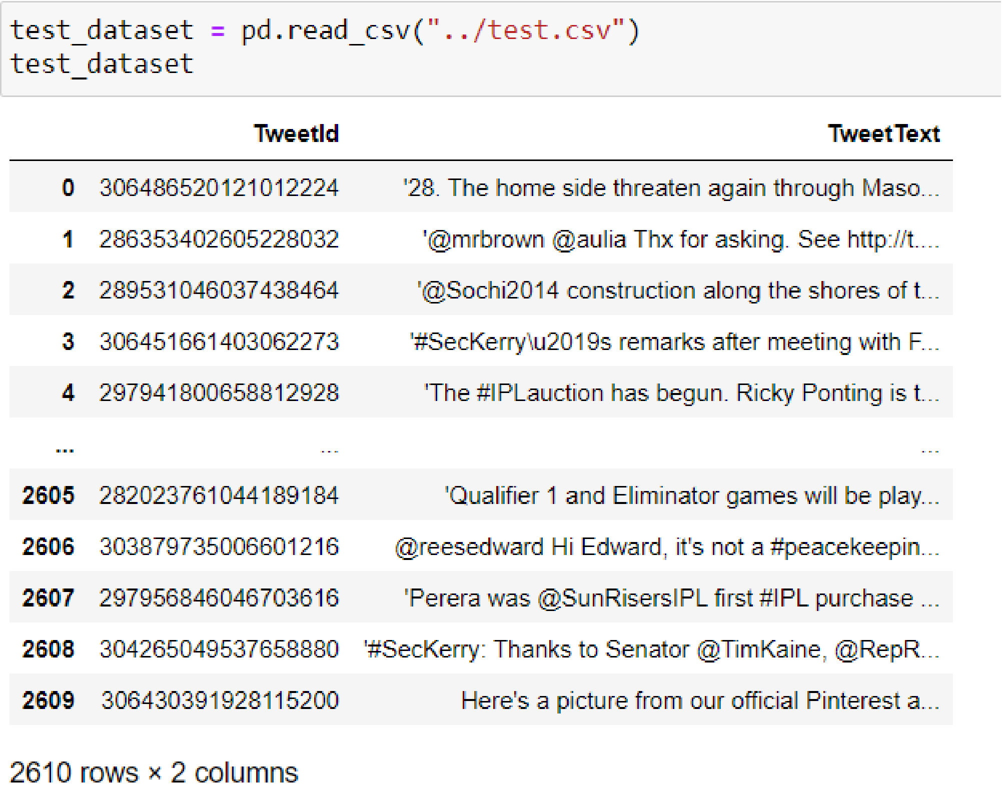
	TweetId
count	6.525000e+03
mean	2.887131e+17
std	5.139819e+16
min	2.390931e+10
25%	2.941380e+17
50%	3.025319e+17
75%	3.053242e+17
max	3.068341e+17



```
train_dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6525 entries, 0 to 6524
Data columns (total 3 columns):
 #   Column     Non-Null Count  Dtype 
---  ------     --------------  ----- 
 0   TweetId    6525 non-null   int64 
 1   Label      6525 non-null   object
 2   TweetText  6525 non-null   object
dtypes: int64(1), object(2)
memory usage: 153.1+ KB
```

The test dataset is composed of two columns ”TweetId”, ”TweetText” and 2610 rows.



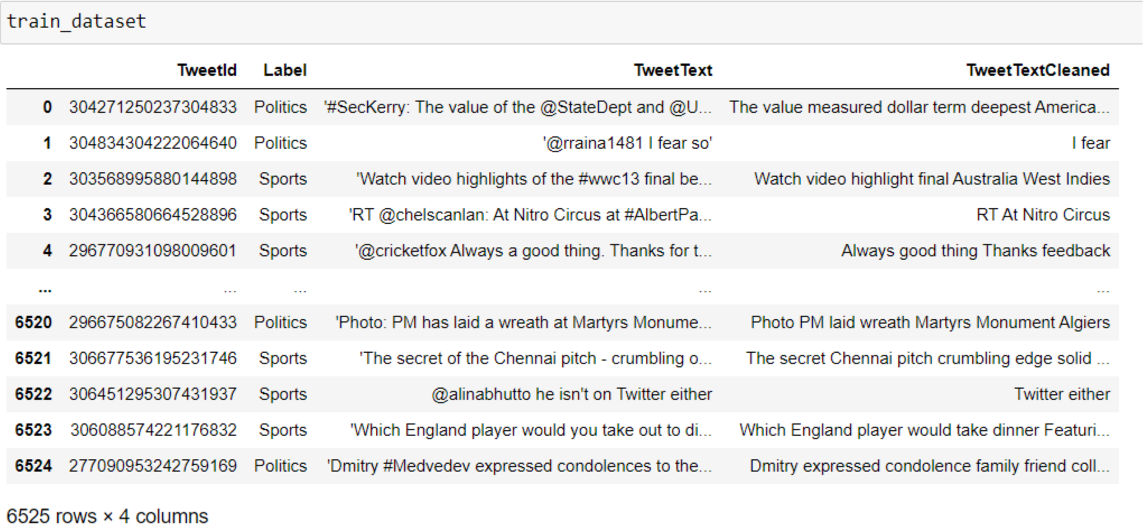
```
test_dataset = pd.read_csv("../test.csv")
test_dataset
```

	TweetId	TweetText
0	306486520121012224	'28. The home side threaten again through Maso...
1	286353402605228032	'@mrbrown @aulia Thx for asking. See http://t....
2	289531046037438464	'@Sochi2014 construction along the shores of t...
3	306451661403062273	'#SecKerry\u2019s remarks after meeting with F...
4	297941800658812928	'The #IPLauction has begun. Ricky Ponting is t...
...	...	...
2605	282023761044189184	'Qualifier 1 and Eliminator games will be play...
2606	303879735006601216	@reesedward Hi Edward, it's not a #peacekeepin...
2607	297956846046703616	'Perera was @SunRisersIPL first #IPL purchase ...
2608	304265049537658880	'#SecKerry: Thanks to Senator @TimKaine, @RepR...
2609	306430391928115200	Here's a picture from our official Pinterest a...

2610 rows × 2 columns

3 Clean dataset :

In this task we have to clean our dastaset by removing urls, hashtags, mentions, characters, puncts, stopwords and doing tokenization and lemmatization.



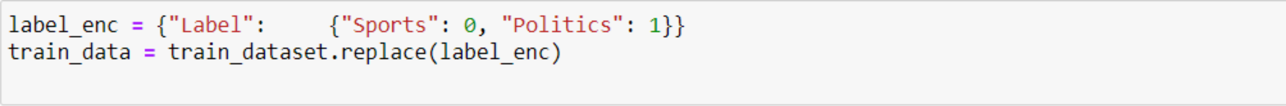
```
train_dataset
```

	TweetId	Label	TweetText	TweetTextCleaned
0	304271250237304833	Politics	'#SecKerry: The value of the @StateDept and @U...	The value measured dollar term deepest America...
1	304834304222064640	Politics	'@rraina1481 I fear so'	I fear
2	303568995880144898	Sports	'Watch video highlights of the #wwc13 final be...	Watch video highlight final Australia West Indies
3	304366580664528896	Sports	'RT @chelscanlan: At Nitro Circus at #AlbertPa...	RT At Nitro Circus
4	296770931098009601	Sports	'@cricketfox Always a good thing. Thanks for t...	Always good thing Thanks feedback
...	...	...	...	...
6520	296675082267410433	Politics	'Photo: PM has laid a wreath at Martyrs Monume...	Photo PM laid wreath Martyrs Monument Algiers
6521	306677536195231746	Sports	'The secret of the Chennai pitch - crumbling o...	The secret Chennai pitch crumbling edge solid ...
6522	306451295307431937	Sports	@alinabhutto he isn't on Twitter either	Twitter either
6523	306088574221176832	Sports	'Which England player would you take out to di...	Which England player would take dinner Featuri...
6524	277090953242759169	Politics	'Dmitry #Medvedev expressed condolences to the...	Dmitry expressed condolence family friend coll...

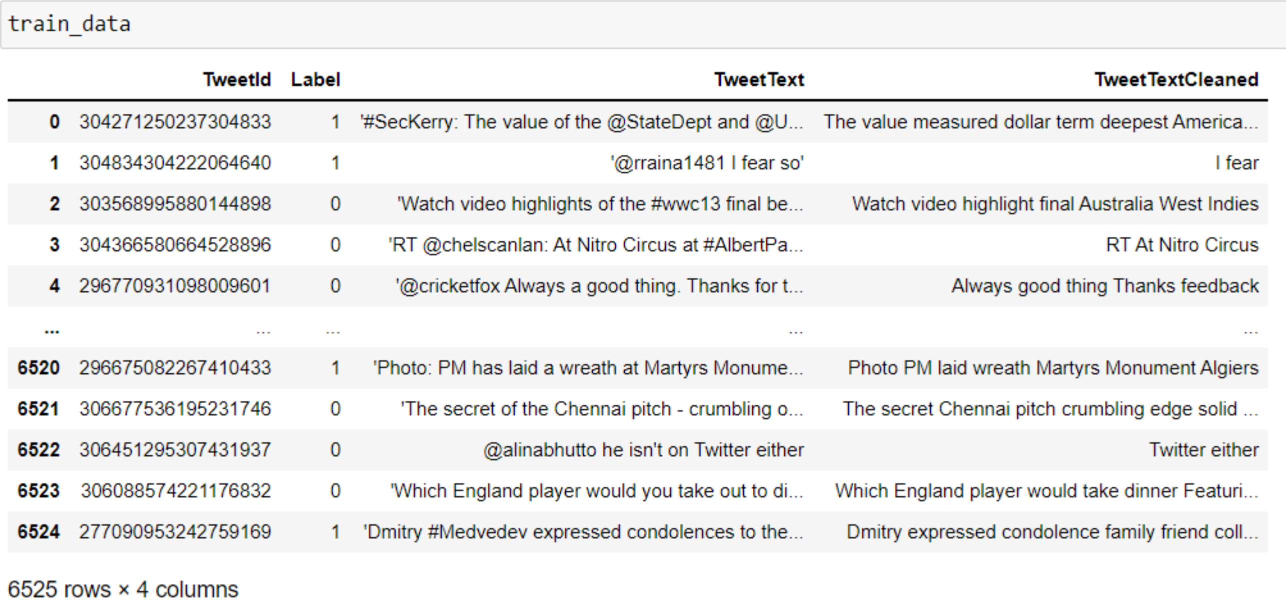
6525 rows × 4 columns

4 Encoding feature label

In this task, we encode the label, Sports 0, Politics 1.



```
label_enc = {"Label":     {"Sports": 0, "Politics": 1}}
train_data = train_dataset.replace(label_enc)
```



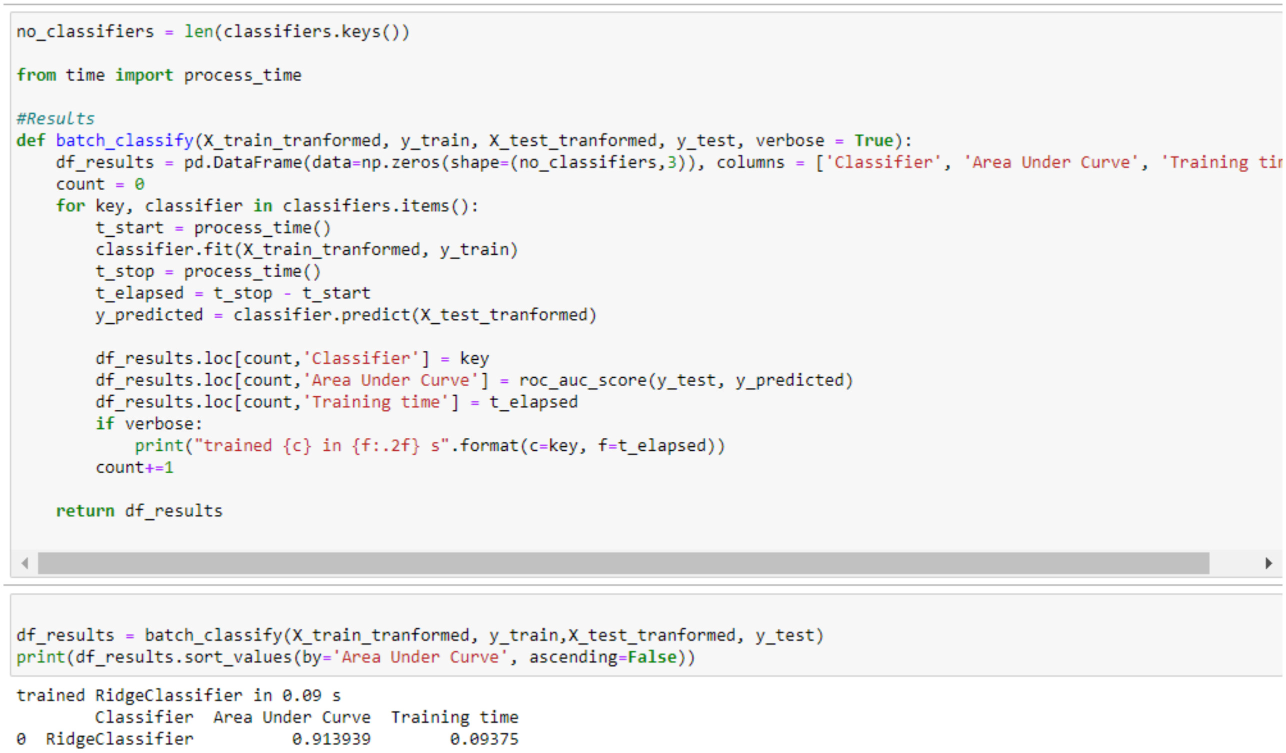
```
train_data
```

	TweetId	Label	TweetText	TweetTextCleaned
0	304271250237304833	1	'#SecKerry: The value of the @StateDept and @U...	The value measured dollar term deepest America...
1	304834304222064640	1	'@rraina1481 I fear so'	I fear
2	303568995880144898	0	'Watch video highlights of the #wwc13 final be...	Watch video highlight final Australia West Indies
3	304366580664528896	0	'RT @chelscanlan: At Nitro Circus at #AlbertPa...	RT At Nitro Circus
4	296770931098009601	0	'@cricketfox Always a good thing. Thanks for t...	Always good thing Thanks feedback
...	...	...	...	...
6520	296675082267410433	1	'Photo: PM has laid a wreath at Martyrs Monume...	Photo PM laid wreath Martyrs Monument Algiers
6521	306677536195231746	0	'The secret of the Chennai pitch - crumbling o...	The secret Chennai pitch crumbling edge solid ...
6522	306451295307431937	0	@alinabhutto he isn't on Twitter either	Twitter either
6523	306088574221176832	0	'Which England player would you take out to di...	Which England player would take dinner Featuri...
6524	277090953242759169	1	'Dmitry #Medvedev expressed condolences to the...	Dmitry expressed condolence family friend coll...

6525 rows × 4 columns

5 Model Evaluation

we choose to work with Ridge classifier, because after evaluation this model, the area under curve is 0.91, and Training time is 0.09 s.



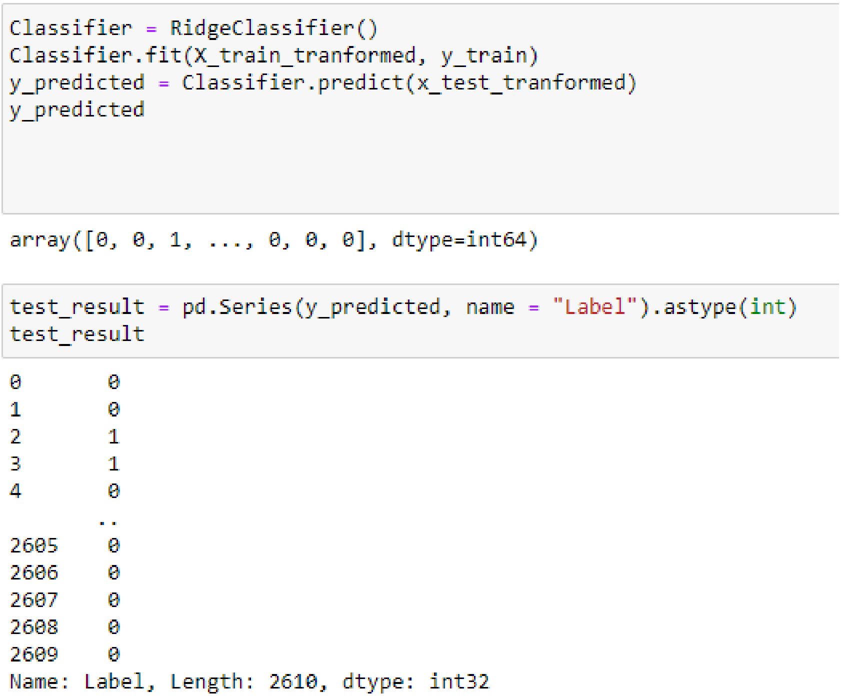
```
no_classifiers = len(classifiers.keys())

from time import process_time

#Results
def batch_classify(X_train_tranformed, y_train, X_test_tranformed, y_test, verbose = True):
    df_results = pd.DataFrame(data=np.zeros(shape=(no_classifiers,3)), columns = ['Classifier', 'Area Under Curve', 'Training ti
    count = 0
    for key, classifier in classifiers.items():
        t_start = process_time()
        classifier.fit(X_train_tranformed, y_train)
        t_stop = process_time()
        t_elapsed = t_stop - t_start
        y_predicted = classifier.predict(X_test_tranformed)

        df_results.loc[count,'Classifier'] = key
        df_results.loc[count,'Area Under Curve'] = roc_auc_score(y_test, y_predicted)
        df_results.loc[count,'Training time'] = t_elapsed
        if verbose:
            print("trained {c} in {f:.2f} s".format(c=key, f=t_elapsed))
        count+=1

    return df_results
```

```
df_results = batch_classify(X_train_tranformed, y_train,X_test_tranformed, y_test)
print(df_results.sort_values(by='Area Under Curve', ascending=False))
```

```
trained RidgeClassifier in 0.09 s
        Classifier  Area Under Curve  Training time
0  RidgeClassifier          0.913939        0.09375
```

6 Predictions

We use this model to predict the category of tweets (Sports or Politics)



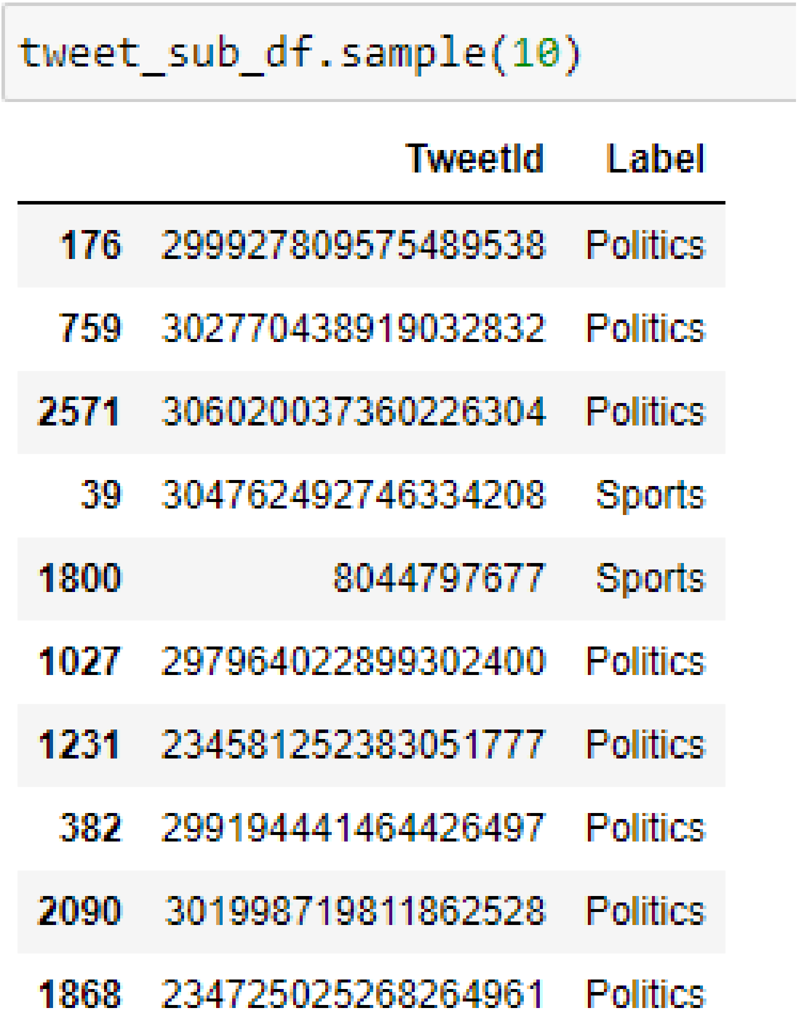
```
Classifier = RidgeClassifier()
Classifier.fit(X_train_tranformed, y_train)
y_predicted = Classifier.predict(x_test_tranformed)
y_predicted
```

```
array([0, 0, 1, ..., 0, 0, 0], dtype=int64)
```

```
test_result = pd.Series(y_predicted, name = "Label").astype(int)
test_result
```

```
0       0
1       0
2       1
3       1
4       0
       ..
2605    0
2606    0
2607    0
2608    0
2609    0
Name: Label, Length: 2610, dtype: int32
```

7 Import results to a csv file



```
tweet_sub_df.sample(10)
```

	TweetId	Label
176	299927809575489538	Politics
759	302770438919032832	Politics
2571	306020037360226304	Politics
39	304762492746334208	Sports
1800	8044797677	Sports
1027	297964022899302400	Politics
1231	234581252383051777	Politics
382	299194441464426497	Politics
2090	301998719811862528	Politics
1868	234725025268264961	Politics