EMAIL SPAM DETECTION PROJECT

Submitted by Nipam Nayan Gogoi

Problem Statement

You were recently hired in start up company and you were asked to build a system to identify spam emails.

It is a Natural Language Processing problem since it is text data.

Data Description

The data is in csv format. It contains attributes: subject, message and label.

Size: 2893 rows and 3 columns in the data.

	A	B	C	D
1	subject	message	label	
2	job posting - apple-iss research center	content - length : 3386 apple-iss research center a us $ 10 million	0	
3		lang classification grimes , joseph e . and barbara f . grimes ;	0	
4	query : letter frequencies for text identification	i am posting this inquiry for sergei atamas (satamas @ umabnet . ab	0	
5	risk	a colleague and i are researching the differing degrees of risk	0	
6	request book information	earlier this morning i was on the phone with a friend of mine living	0	
7	call for abstracts : optimality in syntactic theory	content - length : 4437 call for papers is the best good enough ?	0	
8	m . a . in scandinavian linguistics	m . a . in scandinavian linguistics at the university of tromsoe 1995-	0	
9	call for papers : linguistics session of the m / mla	call for papers linguistics session - - midwest modern language	0	
10	foreign language in commercials	content - length : 1937 greetings ! i ' m wondering if someone out	0	
11	fulbright announcement : please post / disseminate to lists	fulbright announcement : please post / disseminate to lists subject	0	
12	gala ' 95 : call for papers	groningen assembly on language acquisition 1995 university of	0	
13	bu conf on language development ' 95 - announcement	20th annual boston university conference on language development	0	
14	korean software for macintosh	dear sir / madam , would you please send me any information about	0	
15		syntax the antisymmetry of syntax richard s . kayne linguistic inquiry	0	
16	simultaneous prepositions and postpositions in pashto	i ' m looking for analyses of nominal constructions (in any language	0	
17	sum : imperatives without you subjects	content - length : 3573 summary of responses to my query on	0	
18	policies	moderators ' message a very happy 1995 to all our subscribers ! as	0	
19	* * * correction to hellenistic greek announcement	a couple of days ago i send an fyi on hellenistic greek linguistics	0	
20	question on audio samples	i am looking for audio samples of english speech spoken by non-	0	
21	sexism and language	re lydie e . meunier 's latest , i did not mean to say that i consider	0	
22	teaching english in korea	teaching english in korea the language center of the chonnam	0	
23	free	this is a multi-part message in mime format . - - - - - - = _ nextpart _	1	
24	email address for w . dressler	colleagues - we are trying to contact wolfgang dressler of vienna via	0	
25	dhumbadji ! , journal for the history of language	good news for all subscribers , the december issue of dhumbadji !	0	
26	question : quantitative information	hello , there is someone who knows where can i look for "	0	

The predictive analysis is done in Jupyter Notebook. The NLP technique applied is TF-IDF to convert the text data to numerical format for Machine learning application.

tf-idf stands for Term frequency-inverse document frequency. It is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. The tf-idf weight is a weight often used in information retrieval and text mining. Variations of the tf-idf weighting scheme are often used by search engines in scoring and ranking a document's relevance given a query.

Snapshots.

Step1: Importing required packages

In [1]:
```
# Importing required packages
import pandas as pd
import warnings
import numpy as np
from sklearn.feature_extraction.text import CountVectorizer,TfidfVectorizer,TfidfTransformer
from sklearn.ensemble import RandomForestClassifier ,AdaBoostClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import precision_score, recall_score,accuracy_score, classification_report, f1_score ,confusion_matrix
```

In [44]:
```
# Reading the data
data = pd.read_csv("Email Spam detection/messages.csv")
```

In [45]:
```
data
```

Out[45]:

	subject	message	label
0	job posting - apple-iss research center	content - length : 3386 apple-iss research cen...	0
1	NaN	lang classification grimes , joseph e . and ba...	0
2	query : letter frequencies for text identifica...	i am posting this inquiry for sergei atamas (...	0
3	risk	a colleague and i are researching the differin...	0
4	request book information	earlier this morning i was on the phone with a...	0
...	...	...	...
2888	love your profile - ysuolvpv	hello thanks for stopping by ! ! we have taken...	1
2889	you have been asked to join kiddin	the list owner of : " kiddin " has invited you...	1
2890	anglicization of composers ' names	judging from the return post , i must have sou...	0
2891	re : 6 . 797 , comparative method : n - ary co...	gotcha ! there are two separate fallacies in t...	0
2892	re : american - english in australia	hello ! i ' m working on a thesis concerning a	0

In [46]:
```
data.label.value_counts() # Count of Values in Label/ Target column
```

Out[46]:
```
0    2412
1     481
Name: label, dtype: int64
```

In [48]:
```
data.isna().any()          # Their are nan values in the column 'subject'
```

Out[48]:
```
subject     True
message    False
label      False
dtype: bool
```

In [49]:
```
data[data.subject.isna()]
```

Out[49]:

	subject	message	label
1	NaN	lang classification grimes , joseph e . and ba...	0
13	NaN	syntax the antisymmetry of syntax richard s	0
69	NaN	computational ling bengt sigurd (ed) compute...	0
107	NaN	phonology & phonetics burquest , donald a . an...	0
258	NaN	phonology & phonetics leiden in last : hil pho...	0

Step2: Filling the nan values with spaces. Combining the Subject and message column into a single text column for input.

62 rows × 3 columns

In [50]:
```
data = data.fillna(" ") #filling all the NAN values with spaces
```

In [51]:
```
# Combining the subject and messages into single text column which will be input for NLP models later on
data["text"] = data["subject"].astype(str) + " " + data["message"]
```

In [52]:
```
data
```

Out[52]:

	subject	message	label	text
0	job posting - apple-iss research center	content - length : 3386 apple-iss research cen...	0	job posting - apple-iss research center conten...
1		lang classification grimes , joseph e . and ba...	0	lang classification grimes , joseph e . and ...
2	query : letter frequencies for text identifica...	i am posting this inquiry for sergei atamas (...	0	query : letter frequencies for text identifica...
3	risk	a colleague and i are researching the differin...	0	risk a colleague and i are researching the dif...
4	request book information	earlier this morning i was on the phone with a...	0	request book information earlier this morning ...
...	...	...	...	...
2888	love your profile - ysuolvpv	hello thanks for stopping by ! ! we have taken...	1	love your profile - ysuolvpv hello thanks for ...
2889	you have been asked to join kiddin	the list owner of : " kiddin " has invited you...	1	you have been asked to join kiddin the list ow...
2890	anglicization of composers ' names	judging from the return post , i must have sou...	0	anglicization of composers ' names judging fro...
2891	re : 6 . 797 , comparative method : n - ary co...	gotcha ! there are two separate fallacies in t...	0	re : 6 . 797 , comparative method : n - ary co...
2892	re : american - english in australia	hello ! i ' m working on a thesis concerning a...	0	re : american - english in australia hello ! i...

2893 rows × 4 columns

Step3: Cleaning the text

In [53]:
```
# Creating the DF required for the task keeping only the combined column "text" and "label"
nlp_df = data.drop(["subject","message"],axis=1)
```

In [54]:
```
nlp_df
```

Out[54]:

	label	text
0	0	job posting - apple-iss research center conten...
1	0	lang classification grimes , joseph e . and ...
2	0	query : letter frequencies for text identifica...
3	0	risk a colleague and i are researching the dif...
4	0	request book information earlier this morning ...
...	...	...
2888	1	love your profile - ysuolvpv hello thanks for ...
2889	1	you have been asked to join kiddin the list ow...
2890	0	anglicization of composers ' names judging fro...
2891	0	re : 6 . 797 , comparative method : n - ary co...
2892	0	re : american - english in australia hello ! i...

2893 rows × 2 columns

In [55]:
```
# Removing Unnecessary numbers and converting the text into lowercase
nlp_df["text"] = nlp_df["text"].str.lower()
nlp_df["text"] = nlp_df["text"].str.replace('[0-9]','')
#nlp_df["text"] = nlp_df["text"].str.replace('[^\w\s]','')
```

In [55]:
```
# Removing Unnecessary numbers and converting the text into lowercase
nlp_df["text"] = nlp_df["text"].str.lower()
nlp_df["text"] = nlp_df["text"].str.replace('[0-9]','')
#nlp_df["text"] = nlp_df["text"].str.replace('[^\w\s]','')
```

In [56]:
```
nlp_df
```

Out[56]:

	label	text
0	0	job posting - apple-iss research center conten...
1	0	lang classification grimes , joseph e . and ...
2	0	query : letter frequencies for text identifica...
3	0	risk a colleague and i are researching the dif...
4	0	request book information earlier this morning ...
...	...	...
2888	1	love your profile - ysuolvpv hello thanks for ...
2889	1	you have been asked to join kiddin the list ow...
2890	0	anglicization of composers ' names judging fro...
2891	0	re : . , comparative method : n - ary compar...
2892	0	re : american - english in australia hello ! i...

2893 rows × 2 columns

In [58]:
```
y = nlp_df["label"]
X = nlp_df["text"]
```

In [60]:
```
# Splitting into training and Test dataset
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2,random_state=88)
```

Converting into TF-IDF

In [65]:
```
vectorizer = TfidfVectorizer(min_df =1,stop_words='english',use_idf=True,analyzer='word',
                             ngram_range=(1,1),max_features=15000)
x_train = vectorizer.fit_transform(X_train)
x_test  = vectorizer.transform(X_test)
```

tf-idf stands for Term frequency-inverse document frequency. It is a numerical statistic that is intended to reflect how in collection or corpus. The tf-idf weight is a weight often used in information retrieval and text mining. Variations of the tf search engines in scoring and ranking a document's relevance given a query.

In [66]:
```
x_train.shape
```

Out[66]:
```
(2314, 15000)
```

In [67]:
```
x_test.shape
```

Out[67]:
```
(579, 15000)
```

Step4: Converting text data into TF-IDF

Step5: Applying Machine Learning models like Logistic Regression, Random Forest, SVM etc.

Machine Learning Models

Logistic Regression

In [68]:
```
logisticRegr = LogisticRegression(solver='liblinear',class_weight='balanced',random_state=5,tol=0.001,max_iter=1000)
logisticRegr.fit(x_train, y_train)
```

Out[68]:
```
LogisticRegression(class_weight='balanced', max_iter=1000, random_state=5,
                   solver='liblinear', tol=0.001)
```

In [69]:
```
predictions = logisticRegr.predict(x_test)
```

In [70]:
```
cm = confusion_matrix(y_test, predictions)
print(cm)
```

```
[[472   0]
 [  5 102]]
```

In [71]:
```
accuracy_score(y_test, predictions)
```

Out[71]:
```
0.9913644214162349
```

In [72]:
```
print(classification_report(y_test,predictions))
```

```
              precision    recall  f1-score   support

           0       0.99      1.00      0.99       472
           1       1.00      0.95      0.98       107

    accuracy                           0.99       579
   macro avg       0.99      0.98      0.99       579
weighted avg       0.99      0.99      0.99       579
```

Random Forest Model ¶

In [73]:
```
rand = RandomForestClassifier(n_estimators=100,criterion='entropy',max_features=None,class_weight='balanced')
rand.fit(x_train, y_train)
```

Out[73]:
```
RandomForestClassifier(class_weight='balanced', criterion='entropy',
                       max_features=None)
```

In [74]:
```
prediction2 = rand.predict(x_test)
```

In [75]:
```
print('\n','CONFUSION MATRIX','\n',confusion_matrix(y_test, prediction2))
print('\n','ACCURACY','\n',accuracy_score(y_test, prediction2))
print('\n','REPORT','\n',classification_report(y_test,prediction2))
```

```
 CONFUSION MATRIX
 [[468   4]
 [  8  99]]

 ACCURACY
 0.9792746113989638

 REPORT
               precision    recall  f1-score   support

           0       0.98      0.99      0.99       472
           1       0.96      0.93      0.94       107

    accuracy                           0.98       579
   macro avg       0.97      0.96      0.97       579
weighted avg       0.98      0.98      0.98       579
```

SVM

In [76]:
```
from sklearn import svm
SVMM = svm.LinearSVC(class_weight='balanced',verbose=0, random_state=None,max_iter=1000)
```

In [77]:
```
SVMM.fit(x_train,y_train)
predictions3 = SVMM.predict(x_test)
```

In [78]:
```
print('\n','CONFUSION MATRIX','\n',confusion_matrix(y_test, predictions3))
print('\n','ACCURACY','\n',accuracy_score(y_test, predictions3))
print('\n','REPORT','\n',classification_report(y_test,predictions3))
```

```
CONFUSION MATRIX
[[472   0]
 [  7 100]]

ACCURACY
0.9879101899827288

REPORT
              precision    recall  f1-score   support

           0       0.99      1.00      0.99       472
           1       1.00      0.93      0.97       107

    accuracy                           0.99       579
   macro avg       0.99      0.97      0.98       579
weighted avg       0.99      0.99      0.99       579
```

Conclusion

The Logistic Regression is observed as best model for this data to detect spam emails, after converting the text into TF-IDF. This model has accuracy 99% in the test data and also high f1 score, precision and recall. The model is stored as pickle format for deploying later on.

In [79]:
```
#Storing the best model
import joblib

# Saving the model as a pickle in a file
joblib.dump(logisticRegr,"email_spam_prediction.pkl")
```

Out[79]:
```
['email_spam_prediction.pkl']
```

We can use this saved model later on for email spam detection