Spam Mail Detection

P SAIKARTHIK

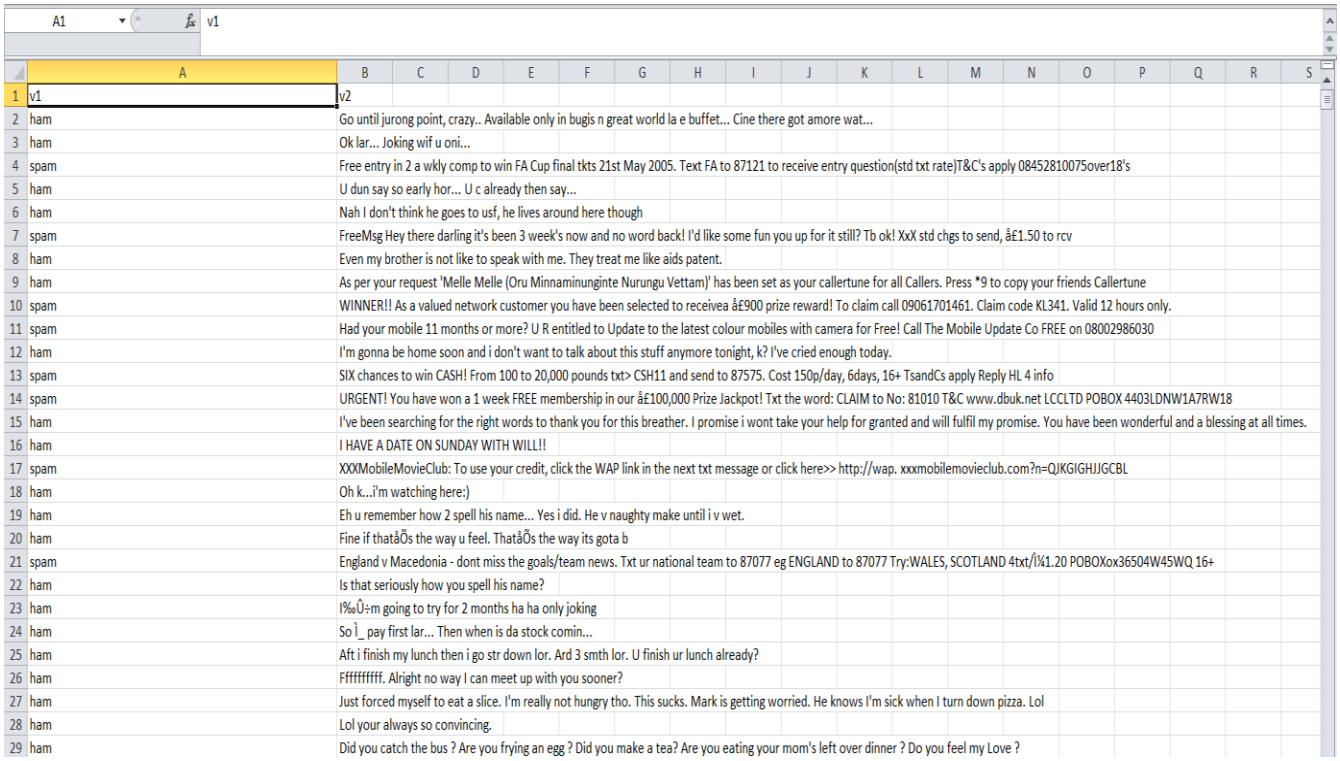
VU21CSEN0100398

1. Introduction

This project involves building a spam email detection system using the Naive Bayes classifier. The goal is to classify emails as spam or non-spam (ham) based on their content.

2. Dataset

The dataset spam.csv is used, which contains two main columns:



	A	B
1	v1	v2
2	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there got amore wat...
3	ham	Ok lar... Joking wif u oni...
4	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive entry question(std txt rate)T&C's apply 08452810075over18's
5	ham	U dun say so early hor... U c already then say...
6	ham	Nah I don't think he goes to usf, he lives around here though
7	spam	FreeMsg Hey there darling it's been 3 week's now and no word back! I'd like some fun you up for it still? Tb ok! XxX std chgs to send, å£1.50 to rcv
8	ham	Even my brother is not like to speak with me. They treat me like aids patent.
9	ham	As per your request 'Melle Melle (Oru Minnaminunginte Nurungu Vettam)' has been set as your callertune for all Callers. Press *9 to copy your friends Callertune
10	spam	WINNER!! As a valued network customer you have been selected to receivea å£900 prize reward! To claim call 09061701461. Claim code KL341. Valid 12 hours only.
11	spam	Had your mobile 11 months or more? U R entitled to Update to the latest colour mobiles with camera for Free! Call The Mobile Update Co FREE on 08002986030
12	ham	I'm gonna be home soon and i don't want to talk about this stuff anymore tonight, k? I've cried enough today.
13	spam	SIX chances to win CASH! From 100 to 20,000 pounds txt> CSH11 and send to 87575. Cost 150p/day, 6days, 16+ TsandCs apply Reply HL 4 info
14	spam	URGENT! You have won a 1 week FREE membership in our å£100,000 Prize Jackpot! Txt the word: CLAIM to No: 81010 T&C www.dbuk.net LCCLTD POBOX 4403LDNW1A7RW18
15	ham	I've been searching for the right words to thank you for this breather. I promise i wont take your help for granted and will fulfil my promise. You have been wonderful and a blessing at all times.
16	ham	I HAVE A DATE ON SUNDAY WITH WILL!!
17	spam	XXXMobileMovieClub: To use your credit, click the WAP link in the next txt message or click here>> http://wap. xxxmobilemovieclub.com?n=QJKGIGHJJGCBL
18	ham	Oh k...i'm watching here:)
19	ham	Eh u remember how 2 spell his name... Yes i did. He v naughty make until i v wet.
20	ham	Fine if thatåÕs the way u feel. thatåÕs the way its gota b
21	spam	England v Macedonia - dont miss the goals/team news. Txt ur national team to 87077 eg ENGLAND to 87077 Try:WALES, SCOTLAND 4txt/Ì¼1.20 POBOXox36504W45WQ 16+
22	ham	Is that seriously how you spell his name?
23	ham	I‰Û÷m going to try for 2 months ha ha only joking
24	ham	So Ì_ pay first lar... Then when is da stock comin...
25	ham	Aft i finish my lunch then i go str down lor. Ard 3 smth lor. U finish ur lunch already?
26	ham	Ffffffffff. Alright no way I can meet up with you sooner?
27	ham	Just forced myself to eat a slice. I'm really not hungry tho. This sucks. Mark is getting worried. He knows I'm sick when I turn down pizza. Lol
28	ham	Lol your always so convincing.
29	ham	Did you catch the bus ? Are you frying an egg ? Did you make a tea? Are you eating your mom's left over dinner ? Do you feel my Love ?

v1: The label indicating if the message is spam ('spam') or ham ('ham').

v2: The message content.

3. Methodology

3.1 Data Preprocessing

Load the dataset and convert message content to lowercase:

python

Copy code

```
df = pd.read_csv('spam.csv', encoding='latin-1')

df['v2'] = df['v2'].str.lower()
```

Define features x (message content) and target y (spam or ham):

python

Copy code

```
x = df['v2']

y = df['v1']
```

3.2 Text Vectorization

Transform text data into numerical features using TfidfVectorizer:

python

Copy code

```
tfi = TfidfVectorizer(stop_words='english')

x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=40)

x_train = tfi.fit_transform(x_train)

x_test = tfi.transform(x_test)
```

3.3 Model Training

Train a MultinomialNB model:

python

Copy code

```
clr = MultinomialNB()

clr.fit(x_train, y_train)
```

3.4 Model Evaluation

Predict on test data and evaluate accuracy:

python

Copy code

y_pred = clr.predict(x_test)

print("Accuracy :", accuracy_score(y_test, y_pred))

print(classification_report(y_test, y_pred))

IMPLEMENTATION

[51]:
```
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import accuracy_score,classification_report

df = pd.read_csv('spam.csv',encoding='latin-1')
df['v2']=df['v2'].str.lower()
x=df['v2']
y=df['v1']
tfi = TfidfVectorizer(stop_words='english')
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.2,random_state=40)
x_train = tfi.fit_transform(x_train)
x_test = tfi.transform(x_test)
clr = MultinomialNB()
clr.fit(x_train,y_train)
y_pred = clr.predict(x_test)
print("Accuracy :",accuracy_score(y_test,y_pred))
print(classification_report(y_test,y_pred))
```

```
Accuracy : 0.9721973094170404
              precision    recall  f1-score   support

         ham       0.97      1.00      0.98       967
        spam       1.00      0.79      0.88       148

    accuracy                           0.97      1115
   macro avg       0.98      0.90      0.93      1115
weighted avg       0.97      0.97      0.97      1115
```

[11]:
```
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score,classification_report
```

[9]:
```
df = pd.read_csv('spam.csv',encoding='latin-1')
df['v2']=df['v2'].str.lower()
x = df['v2']
y = df['v1']
tfi = TfidfVectorizer(stop_words='english')
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.2,random_state=40)
x_train = tfi.fit_transform(x_train)
x_test = tfi.transform(x_test)

clr = LogisticRegression()
clr.fit(x_train,y_train)
```

[9]:
```
LogisticRegression()
```

[ ]:
```
##method 3
```

[27]:
```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.svm import SVC
from sklearn.metrics import accuracy_score,classification_report
```

[26]:
```
df=pd.read_csv('spam.csv',encoding = 'latin-1')
df['v2']=df['v2'].str.lower()
x = df['v2']
y= df['v1']
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.2,random_state=40)
tfi = TfidfVectorizer(stop_words='english')
x_train = tfi.fit_transform(x_train)
x_test = tfi.transform(x_test)

clr = SVC(kernel='linear')
clr.fit(x_train,y_train)
```

[26]:
```
SVC(kernel='linear')
```

4. Results

Accuracy: The accuracy of the model on the test set is displayed.

Classification Report: Provides precision, recall, and F1-score for each class (spam and ham).

```
l="WINNER!! As a valued network customer you have been selected to receivea å£900 prize reward! To claim call 09061701461. Claim code KL341. Valid 12 hou
l=l.lower()
l= tfi.transform([l])
y_pred = clr.predict(l)
print(y_pred[0])
```

```
spam
```

```
[13]: y_pred = clr.predict(x_test)
print("accuracy :",accuracy_score(y_test,y_pred))
print(classification_report(y_test,y_pred))
l="WINNER!! As a valued network customer you have been selected to receivea å£900 prize reward! To claim call 09061701461. Claim code KL341. Valid 12 hou
l=l.lower()
l= tfi.transform([l])
y_pred = clr.predict(l)
print(y_pred[0])
```

```
accuracy : 0.9650224215246637
              precision    recall  f1-score   support

         ham       0.96      1.00      0.98       967
        spam       0.99      0.74      0.85       148

    accuracy                           0.97      1115
   macro avg       0.98      0.87      0.91      1115
weighted avg       0.97      0.97      0.96      1115

spam
```

```
[29]: y_pred = clr.predict(x_test)
print("accuracy : ",accuracy_score(y_test,y_pred))
print(classification_report(y_test,y_pred))
l="WINNER!! As a valued network customer you have been selected to receivea å£900 prize reward! To claim call 09061701461. Claim code KL341. Valid 12 hou
l=l.lower()
l=tfi.transform([l])
l_pred = clr.predict(l)
print(l_pred[0])
```

```
accuracy :  0.9865470852017937
              precision    recall  f1-score   support

         ham       0.99      1.00      0.99       967
        spam       0.99      0.91      0.95       148

    accuracy                           0.99      1115
   macro avg       0.99      0.96      0.97      1115
weighted avg       0.99      0.99      0.99      1115

spam
```

5. Prediction Example

Classify a sample message:

python

Copy code

l = "WINNER!! As a valued network customer you have been selected to receivea å£900 prize reward! To claim call 09061701461. Claim code KL341. Valid 12 hours only."

l = l.lower()

l = tfi.transform([l])

y_pred = clr.predict(l)

print(y_pred[0])

This sample message is classified as spam by the model.

6. Conclusion

The Naive Bayes classifier effectively distinguishes between spam and non-spam emails. The model's performance is evaluated using accuracy and a detailed classification report, demonstrating its capability in spam detection.

🌐 Sources

github.com - Spam Email Detection Using MultinomialNB

analyticsvidhya.com - End-to-End Project on SMS/Email Spam Detection using Naive Bayes

towardsdatascience.com - Naïve Bayes Spam Filter — From Scratch

kaggle.com - Naïve Bayes Classification : Spam Email Detection

github.com - SMS spam detection by using the Naive Bayes

researchgate.net - Spam Detection in Twitter Using Multinomial Naive Bayes Classifier