EMAIL SPAM DETECTOR

NITIN I AND SAURAV KUMAR YADAV

DATA SET USED: 'spam.csv' from Kaggle.com

Email_Spam_Model.ipynb:

In [66]:
```
import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import confusion_matrix, plot_confusion_matrix
import matplotlib.pyplot as plt
import pickle
```

In [31]:
```
data = pd.read_csv("spam.csv", encoding="latin-1")
```

In [32]:
```
data.head(5)
```

Out[32]:

	class	message	Unnamed: 2	Unnamed: 3	Unnamed: 4
0	ham	Go until jurong point, crazy.. Available only ...	NaN	NaN	NaN
1	ham	Ok lar... Joking wif u oni...	NaN	NaN	NaN
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...	NaN	NaN	NaN
3	ham	U dun say so early hor... U c already then say...	NaN	NaN	NaN
4	ham	Nah I don't think he goes to usf, he lives aro...	NaN	NaN	NaN

In [33]:
```
data.columns
```

Out[33]:
```
Index(['class', 'message', 'Unnamed: 2', 'Unnamed: 3', 'Unnamed: 4'], dtype='object')
```

In [34]:
```
data.drop(['Unnamed: 2', 'Unnamed: 3', 'Unnamed: 4'], axis = 1, inplace = True)
```

In [35]:
```
data.head()
```

Out[35]:

	class	message
0	ham	Go until jurong point, crazy.. Available only ...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives aro...

In [36]:
```
data['class'] = data['class'].map({'ham':0,'spam':1})
cv = CountVectorizer()
x = data['message']
y = data['class']
```

In [40]:
```
x.shape
```

Out[40]:
```
(5572,)
```

In [41]:
```
y.shape
```

Out[41]:
```
(5572,)
```

In [42]:
```
x = cv.fit_transform(x)
```

In [43]:
```
x
```

Out[43]:
```
<5572x8672 sparse matrix of type '<class 'numpy.int64'>'
        with 73916 stored elements in Compressed Sparse Row format>
```

In [44]:
```
x_train, x_test, y_train, y_test = train_test_split( x, y, test_size = 0.2)
```

In [54]:
```
x_train.shape
```

Out[54]:
```
(4457, 8672)
```

In [55]:
```
y_train
```

Out[55]:
```
114     0
1420    0
5431    0
2964    0
1026    0
       ..
2009    0
4589    0
3733    0
4018    1
3082    0
Name: class, Length: 4457, dtype: int64
```

In [47]:
```
model = MultinomialNB()
```

In [48]:
```
model.fit(x_train,y_train)
```

Out[48]:
```
MultinomialNB()
```

In [49]:
```
model.score(x_test,y_test)
```

Out[49]:
```
0.9775784753363229
```

In [50]:
```
pickle.dump( model, open("spam.pkl","wb"))
```

In [51]:
```
pickle.dump( cv, open("vectorizer.pkl","wb"))
```

In [52]:
```
msg = "free camcorder"
data = [msg]
vect = cv.transform(data).toarray()
result = model.predict(vect)
print(result)
```

```
[1]
```

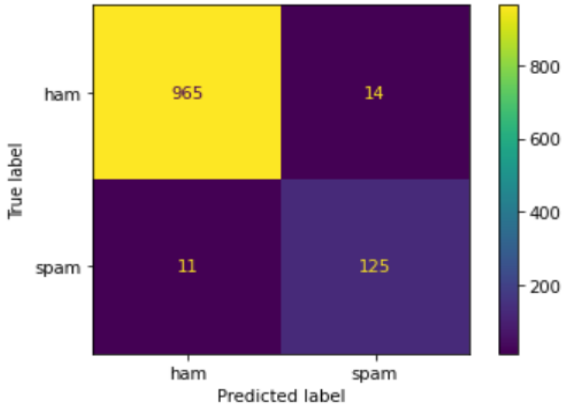
In [60]:
```
y_pred = model.predict(x_test)
```

In [68]:
```
confusion_matrix(y_test,y_pred)
```

Out[68]:
```
array([[965,  14],
       [ 11, 125]], dtype=int64)
```

In [67]:
```
plot_confusion_matrix(model,x_test.toarray(),y_test,display_labels=['ham','spam'])
```

Out[67]:
```
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x2b4cd1f8df0>
```

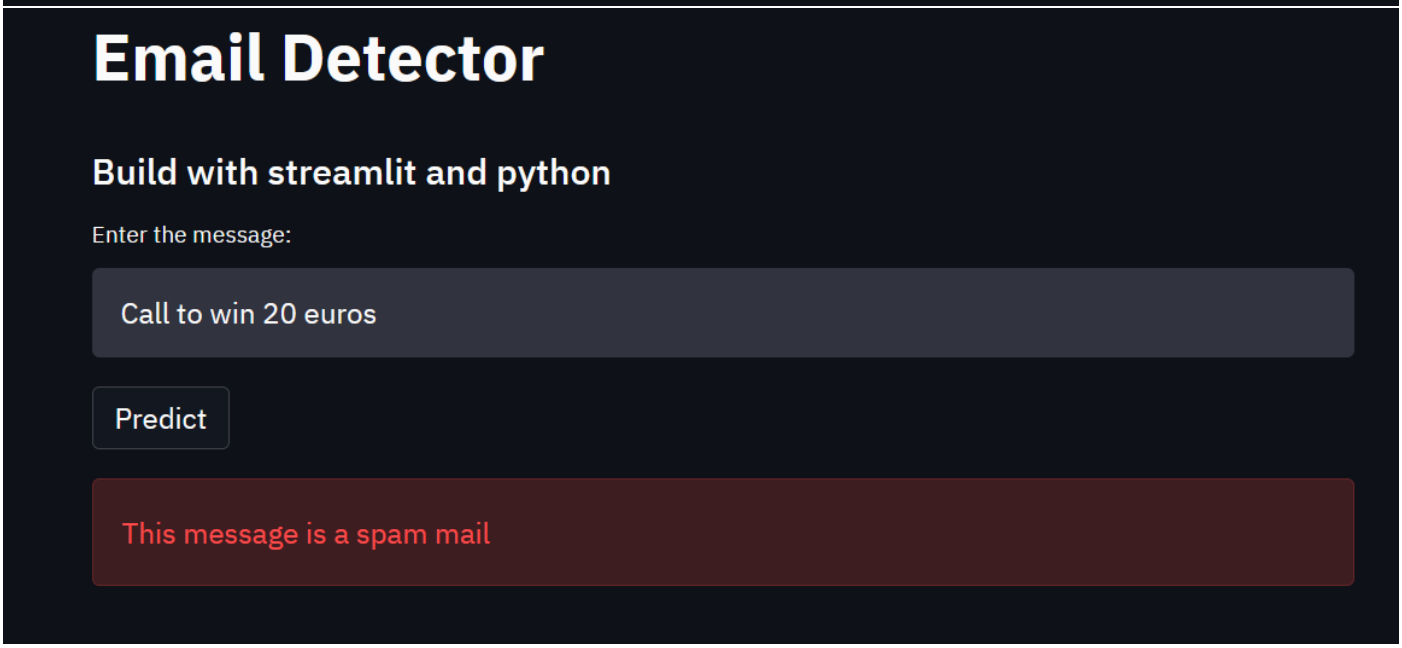


spamDetector.py:

```python
import pickle
import streamlit as st

model = pickle.load(open("spam.pkl","rb"))
cv = pickle.load(open("vectorizer.pkl","rb"))

def main():
    st.title("Email Detector")
    st.subheader("Build with streamlit and python")
    msg = st.text_input("Enter the message:")
    if st.button("Predict"):
        data = [msg]
        vect = cv.transform(data).toarray()
        prediction = model.predict(vect)
        result = prediction[0]
        if result == 1:
            st.error("This message is a spam mail")
        else:
            st.success("This message is a ham mail")

main()
```

Output:


Email Detector
Build with streamlit and python
Enter the message:
Hello Greg how are you
Predict
This message is a ham mail


Email Detector
Build with streamlit and python
Enter the message:
Call to win 20 euros
Predict
This message is a spam mail