EMAIL/SMS SPAM DETECTION

A project report submitted in partial fulfillment of the requirement for the degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE & ENGINEERING

By

B.POOJITHA(R170621)

Under the guidance of

Ms C. SUNEETHA
Asst.Professor
Department of CSE

AP IIIT, RGUKT-RK Valley

Vempalli, Kadapa (Dist), Andhra Pradesh-516330,India

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES

(A.P.Government Act 18 of 2008)
RGUKT-RK Valley
Vempalli, Kadapa, Andhrapradesh-516330.

CERTIFICATE OF PROJECT COMPLETION

This is to certify that I have examined the thesis entitled submitted by G.Sneha(R170614) and B.poojitha(R170621) under our guidance and supervision for the partial fulfillment for the degree of Bachelor of Technology in Computer Science and Engineering during the academic session September 2022 – April 2023 at RGUKT - RK Valley. To the best of my knowledge, the results embodied in this dissertation work have not been submitted to any university or institute for the award of any degree or diploma.

Project Internal Guide

Ms C. Suneetha
Asst.Professor,
RGUKT,RK Valley.

Head of the Department

Mr. N. Satyanandaram
HOD of CSE,
RGUKT,RK Valley.

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES

(A.P.Government Act 18 of 2008)
RGUKT-RK Valley
Vempalli, Kadapa, Andhrapradesh-516330.

DECLARATION

We , G.Sneha(R170614) and B.Poojitha(R170621) hereby declare that the project report entitled “Email/Sms spam detection” done by is under the guidance of Ms C. Suneetha is submitted in partial fulfillment for the degree of Bachelor of Technology in Computer Science and Engineering during the academic session September 2022 – April 2023 at RGUKT-RK Valley. I also declare that this project is a result of our own effort and has not been copied or imitated from any source. Citations from any websites are mentioned in the references. To the best of my knowledge, the results embodied in this dissertation work have not been submitted to any university or institute for the award of any degree or diploma.

G.Sneha(R170614)

B.Poojitha(R170621)

ACKNOWLEDGEMENT

We would like to express our deep sense of gratitude & respect to all those people behind the screen who guided, inspired and helped us crown all our efforts with success. I wish to express our gratitude to our project guide Ms. C. Suneetha for her valuable guidance at all stages of study, advice, constructive suggestions, supportive attitude and continuous encouragement, without which it would not be possible to complete this project.

We would also like to extend our deepest gratitude & reverence to the HOD of the computer science and engineering **Mr.N. Satyanandaram** RGUKT,RK Valley and the Director of RGUKT, **Prof. K. Sandhyarani** for their constant support and encouragement.

Last but not least I express my gratitude to my parents for their constant source of encouragement and inspiration for me to keep my morals high.

Table of Contents Page No.

Abstract

Nowadays, a big part of people rely on available email or messages sent by the stranger.The possibility that email or messages sent by the stranger. The possibility that anybody can leave an email or message provides a golden opportunity for spammers to write spam message about different interests .Spam fills inbox with number of ridiculous emails. Degrades our internet speed to a great extent .Steals useful information like our details on our contact list. Identifying these spammers and also the spam content can be a hot topic of research and laborious tasks. Email spam is an operation to send messages in bulk by mail .Since the expense of the spam is borne mostly by the recipient ,it is effectively postage due advertising. Spam email is a kind of commercial advertising which is economically viable because email could be a very cost effective medium for sender .With this proposed model the specified message can be stated as spam or not using Bayes' theorem and Naive Bayes' Classifier and Also IP addresses of thesender are often detected.

Problem Statement

In this system, to solve the problem of spam, the spam classification system is created to identify spam and non- spam. Since spammers may send spam messages many times, it is difficult to identify it every time manually .So we will be using some of the strategies in our proposed system to detect the spam. The proposed solution not only identifies the spam word but also identifies the IP address of the system through which the spam message is sent so that next time when the spam message is sent from the same system our proposed system directly identifies it as blacklisted based on the IP address.In the proposed model ,the web application is done using dot net and spam detection is done using machine learning.

Introduction

In recent years, internet has become an integral part of life.With increased use of internet, numbers of email users are increasing day by day. This increasing use of email has created problems caused by unsolicited bulk email messages commonly referred to as Spam. Email has now become one of the best ways for advertisements due to which spam emails are generated. Spam emails are the emails that the receiver does not wish to receive. a large number of identical messages are sent to several recipients of email. Spam usually arises as a result of giving out our email address on an unauthorized or unscrupulous website .There are many of the effects of Spam .Fills our Inbox with number of ridiculous emails .Degrades our Internet speed to a great extent .Steals useful information like our details on you Contact list .Alters your search results on any computer program .Spam is a huge waste of everybody's time and can quickly become very frustrating if you receive large amounts of it .Identifying these spammers and the spam content is a laborious task . even though extensive number of studies have been done, yet so far the methods set forth still scarcely distinguish spam surveys, and none of them demonstrate the benefits of each removed element compose .In spite of increasing network communication and wasting lot of memory space ,spam messages are also used for some attack . Spam emails, also known as non-self, are unsolicited commercial or malicious emails, sent to affect either a single individual or a corporation or a bunch of people. Besides advertising, these may contain links to phishing or malware hosting websites found out to steal confidential information. to solve this problem the different spam filtering techniques are used. The spam filtering techniques are used.

Existing System

Existing for Email/Sms spam detection mainly rely on ruled based approaches these approaches have limitations in terms of accuracy and scalability.Machine learning techniques have been shown to be more effective in detecting such contents.

Proposed System

The proposed system used a decision tree classifier to classify whether the message is spam or not.In this Exploratory data analysis process is used to preprocessing by removing stop words and punctuations.The remaining words are stemmed and lemmitized using snow ball stremmer algorithm.A count vectorizer is used to create a matrix of word frequencies which is then fed into the decision classifier for training and testing.

Technologies Used:

Front-end:

- Python
- Streamlit

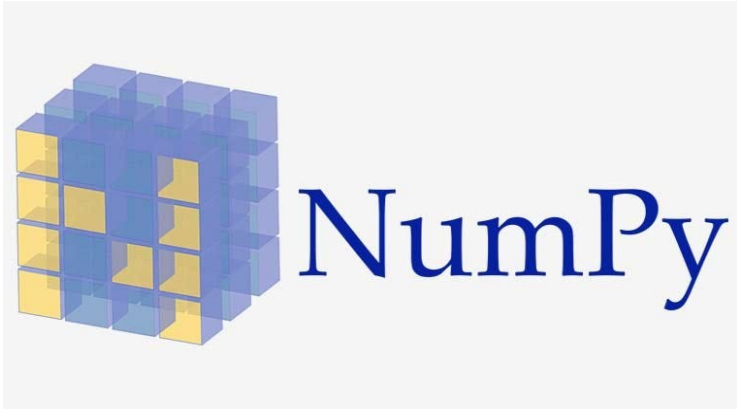
Back-end:

- Python

Libraries used:

- Python pandas
- NUMPY
- Scikit learn
- NLTK(Natural language tool kit)


Pandas


NumPy


scikit
learn


python
NLTK

Modules:

→ Stopwords

→ Stemming

→ Tokenization

1.Stopwords

The words which are generally filtered out before processing a natural language are called **stop words**. These are actually the most common words in any language (like articles, prepositions, pronouns, conjunctions, etc) and does not add much information to the text. Examples of a few stop words in English are “the”, “a”, “an”, “so”, “what”.

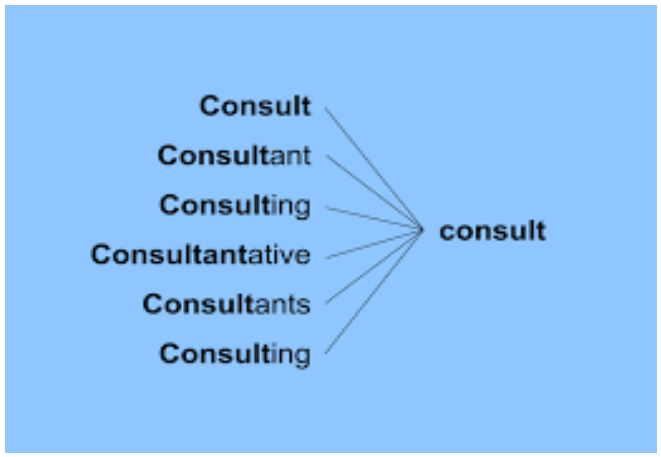
Example:-

Sample text with Stop Words	Without Stop Words
GeeksforGeeks – A Computer Science Portal for Geeks	GeeksforGeeks , Computer Science, Portal ,Geeks
Can listening be exhausting?	Listening, Exhausting
I like reading, so I read	Like, Reading, read

2.Stemming

Stemming is a technique used to extract the base form of the words by removing affixes from them.It is just like cutting down the branches of a tree to its stems

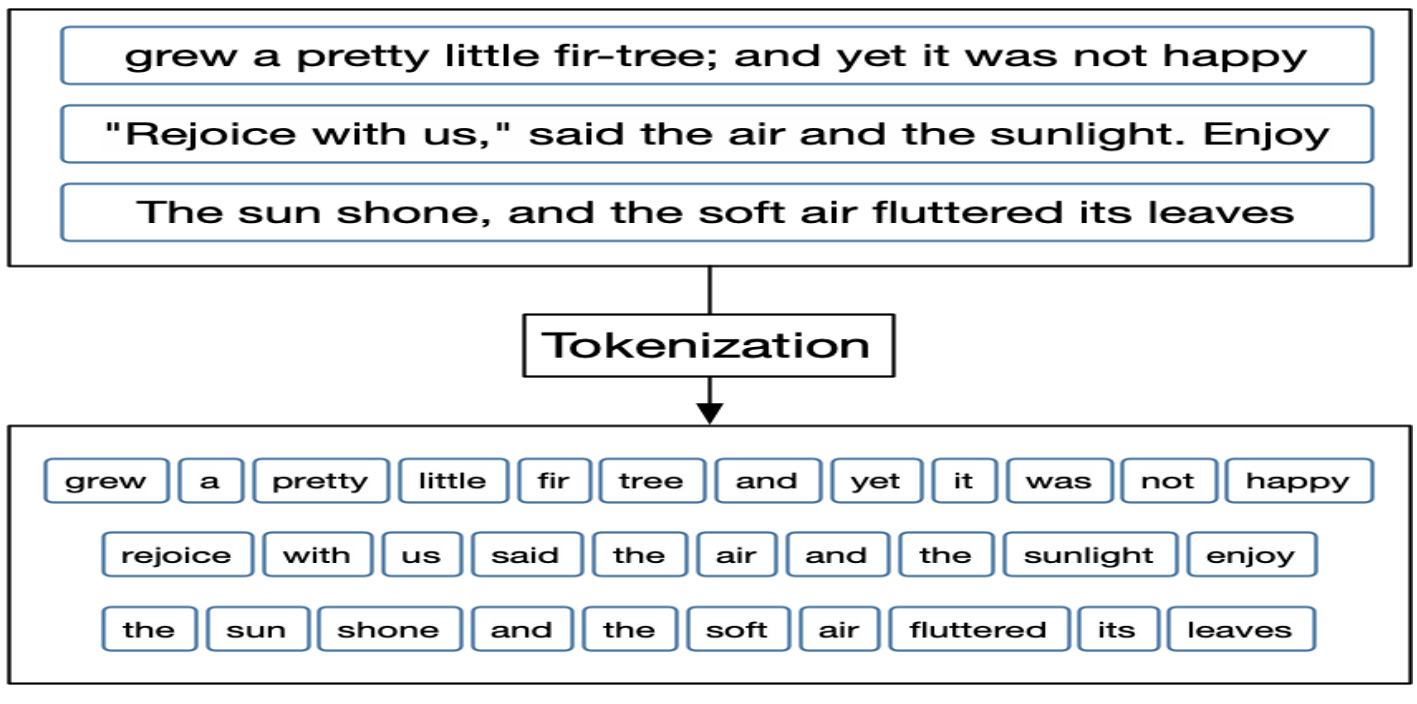
Example:



3.Tokenization

Tokenization is the process of breaking a stream of text up into words,phrases,symbols,or other meaningful elements called Tokens.

Example:



Methods:

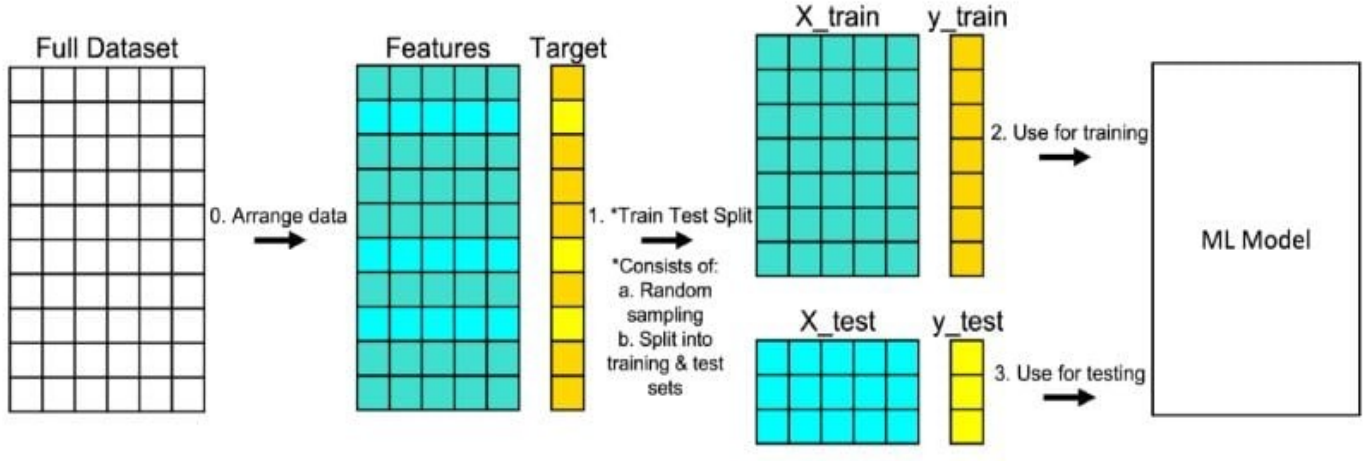
➔ Count vectoriser()

CountVectorizer means breaking down a sentence or any text into words by performing preprocessing tasks like converting all words to lowercase, thus removing special characters. In NLP models can't understand textual data they only accept numbers, so this textual data needs to be vectorized

	Hello	James	hello	is	james	my	name
0	0	0	1	1	1	1	1
1	1	1	0	1	0	1	1

➔ train_test_split()

A train test split is when you split your data into a training set and a testing set. The training set is used for training the model, and the testing set is used to test your model. This allows you to train your models on the training set, and thentest their accuracy on the unseen testing set.



Source Code

Front-end

app.py

C: > Users > agdha > OneDrive > Desktop > sms-spam-classifier-main > app.py

```python
import streamlit as st
import pickle
import string
from nltk.corpus import stopwords
import nltk
from nltk.stem.porter import PorterStemmer


ps = PorterStemmer()


def transform_text(text):
    text = text.lower()
    text = nltk.word_tokenize(text)

    y = []
    for i in text:
        if i.isalnum():
            y.append(i)

    text = y[:]
    y.clear()

    for i in text:
        if i not in stopwords.words('english') and i not in string.punctuation:
            y.append(i)

    text = y[:]
    y.clear()

    for i in text:
        y.append(ps.stem(i))

    return " ".join(y)

tfidf = pickle.load(open('C:/Users/agdha/OneDrive/Desktop/sms-spam-classifier-main/vectorizer.pkl','rb'))
model = pickle.load(open('C:/Users/agdha/OneDrive/Desktop/sms-spam-classifier-main/model.pkl','rb'))

st.title("Email/SMS Spam Classifier")

input_sms = st.text_area("Enter the message")

if st.button('Predict'):

    # 1. preprocess
    transformed_sms = transform_text(input_sms)
    # 2. vectorize
    vector_input = tfidf.transform([transformed_sms])
    # 3. predict
    result = model.predict(vector_input)[0]
    # 4. Display
    if result == 1:
        st.header("Spam")
    else:
        st.header("Not Spam")
```

Back-end:

C: > Users > agdha > OneDrive > Desktop > sms-spam-classifier-main > sms_spam_detection.ipynb > import numpy as np

+ Code + Markdown ··· Select Kernel

1. Data Cleaning

```
df.info()
```

[10] Python

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5572 entries, 0 to 5571
Data columns (total 5 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   v1          5572 non-null   object
 1   v2          5572 non-null   object
 2   Unnamed: 2  50 non-null     object
 3   Unnamed: 3  12 non-null     object
 4   Unnamed: 4  6 non-null      object
dtypes: object(5)
memory usage: 217.8+ KB
```

```
# drop last 3 cols
#df.drop(columns=['Unnamed: 2','Unnamed: 3','Unnamed: 4'],inplace=True)
```

[12] Python

+ Code + Markdown ··· Select Kernel

```
import numpy as np
import pandas as pd
import nltk
nltk.download('punkt')
nltk.download('stopwords')
```

[6] Python

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\agdha\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\agdha\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!

True
```

Python

```
df = pd.read_csv(r'C:\Users\agdha\OneDrive\Desktop\sms-spam-classifier-main\spam.csv',encoding='ISO-8859-1')
```

[7] Python

C: > Users > agdha > OneDrive > Desktop > sms-spam-classifier-main > sms_spam_detection.ipynb > import numpy as np

+ Code + Markdown ··· Select Kernel

2.EDA

```
df.head()
```
[23] Python

```
df['target'].value_counts()
```
[24] Python

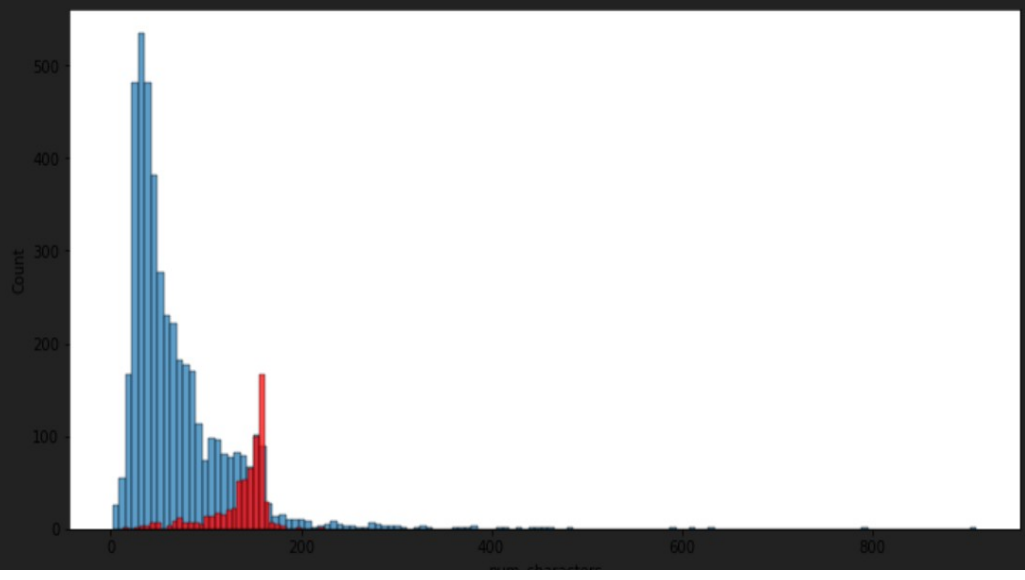
```
0    4516
1     653
Name: target, dtype: int64
```

```
import matplotlib.pyplot as plt
plt.pie(df['target'].value_counts(), labels=['ham','spam'],autopct="%0.2f")
plt.show()
```
[26] Python

+ Code + Markdown ··· Select Kernel

```
plt.figure(figsize=(12,6))
sns.histplot(df[df['target'] == 0]['num_characters'])
sns.histplot(df[df['target'] == 1]['num_characters'],color='red')
```
[44] Python

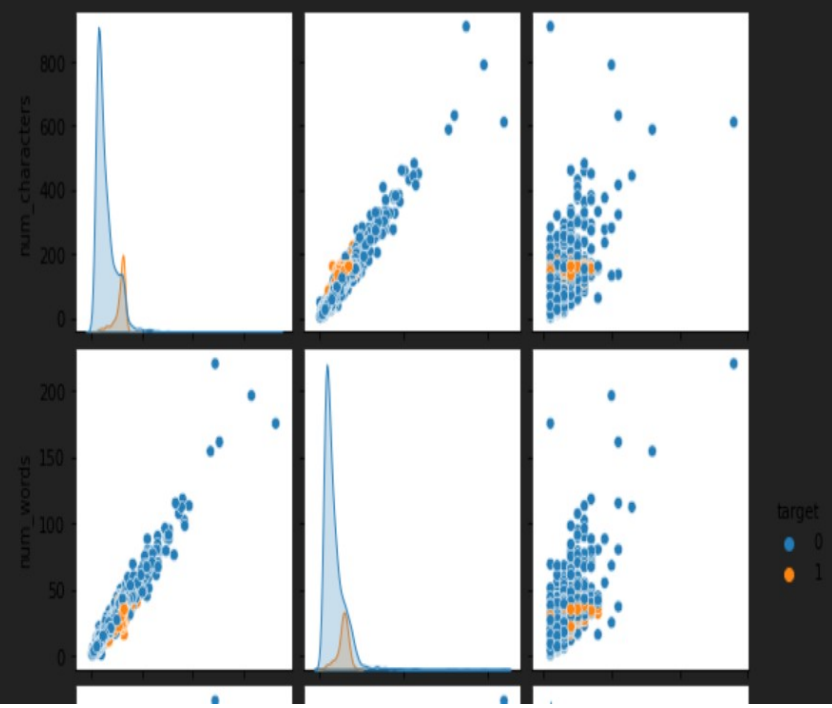
```
<AxesSubplot:xlabel='num_characters', ylabel='Count'>
```



```
sns.pairplot(df,hue='target')
```

[46] Python

```
<seaborn.axisgrid.PairGrid at 0x1fd9dfb9e80>
```



3. Data Preprocessing

- Lower case
- Tokenization
- Removing special characters
- Removing stop words and punctuation
- Stemming

```
from nltk.corpus import stopwords
import string
```

[67] Python

```
def transform_text(text):
    text = text.lower()
    text = nltk.word_tokenize(text)

    y = []
    for i in text:
        if i.isalnum():
            y.append(i)
```

C: > Users > agdha > OneDrive > Desktop > sms-spam-classifier-main > sms_spam_detection.ipynb > import numpy as np

+ Code + Markdown ··· Select Kernel

```
ham_wc = wc.generate(df[df['target'] == 0]['transformed_text'].str.cat(sep=" "))
```

Python

```
plt.figure(figsize=(15,6))
plt.imshow(ham_wc)
```

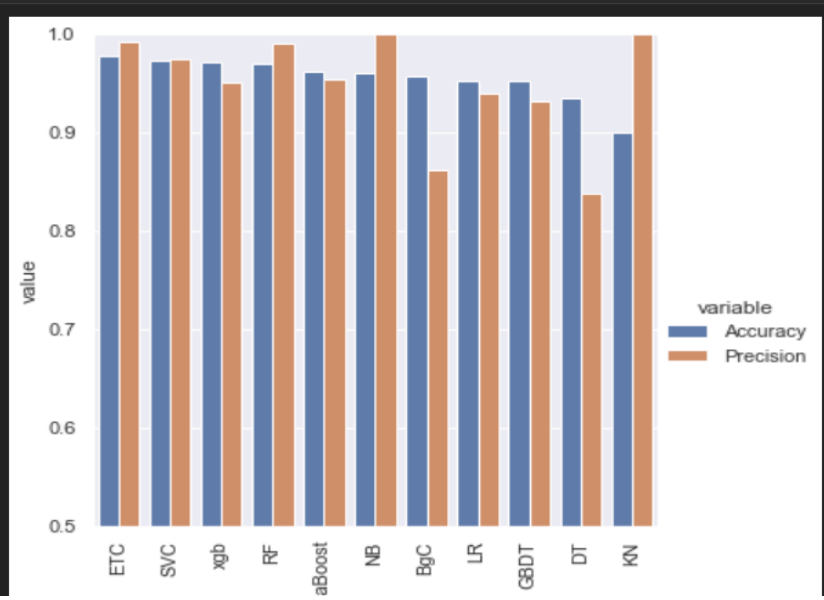
Python

```
<matplotlib.image.AxesImage at 0x16f87f6c280>
```

C: > Users > agdha > OneDrive > Desktop > sms-spam-classifier-main > sms_spam_detection.ipynb > Model Building > bnb.fit(X_train,y_train)

+ Code + Markdown ···

```
sns.catplot(x = 'Algorithm', y='value',
                    hue = 'variable',data=performance_df1, kind='bar',height=5)
plt.ylim(0.5,1.0)
plt.xticks(rotation='vertical')
plt.show()
```



C: > Users > agdha > OneDrive > Desktop > sms-spam-classifier-main > sms_spam_detection.ipynb > M↓Model Building > bnb.fit(X_train,y_train)

+ Code + Markdown ··· Select Kern

```
from sklearn.ensemble import StackingClassifier
```

Pythor

```
clf = StackingClassifier(estimators=estimators, final_estimator=final_estimator)
```

Pythor

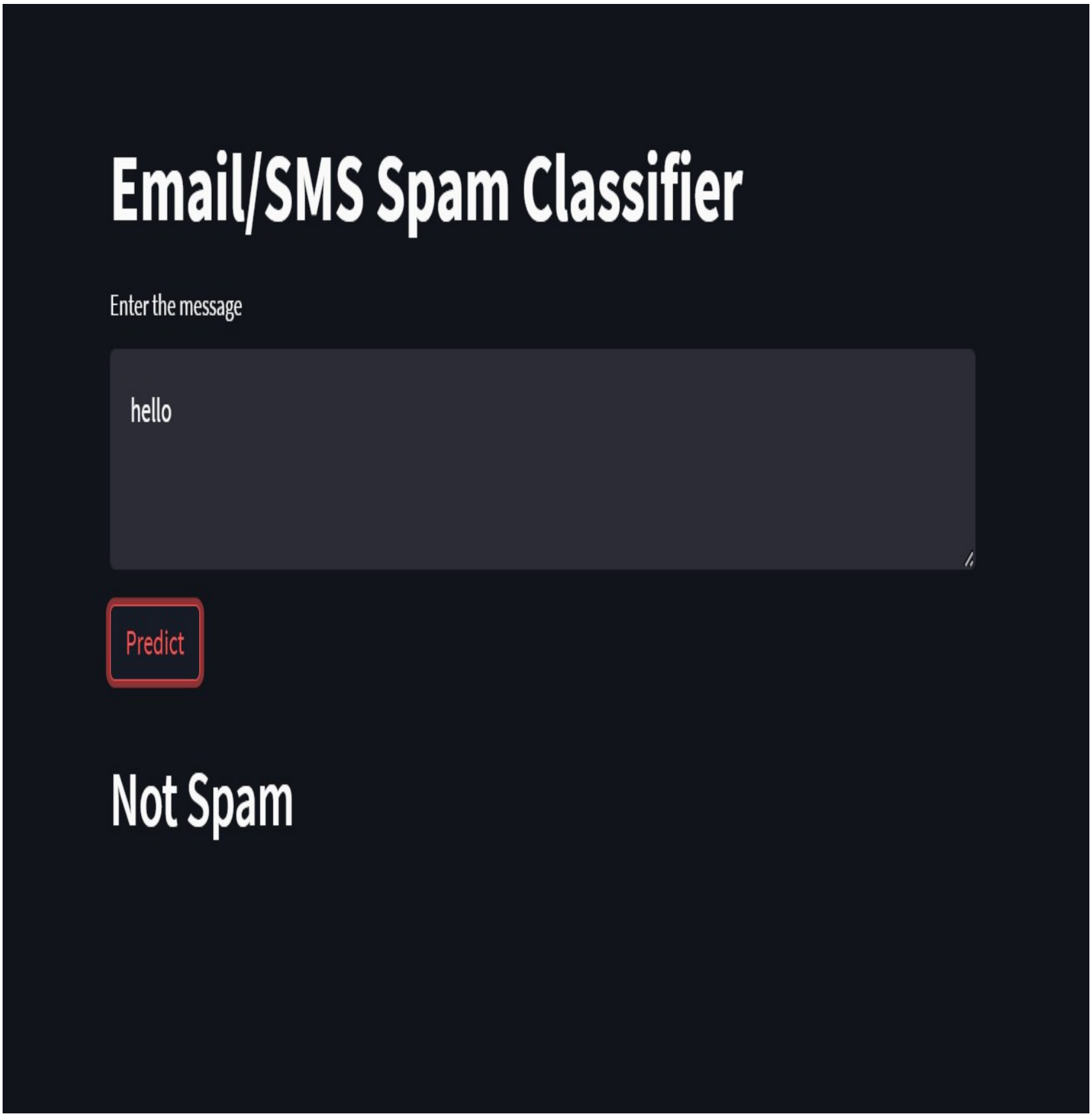
```
clf.fit(X_train,y_train)
y_pred = clf.predict(X_test)
print("Accuracy",accuracy_score(y_test,y_pred))
print("Precision",precision_score(y_test,y_pred))
```

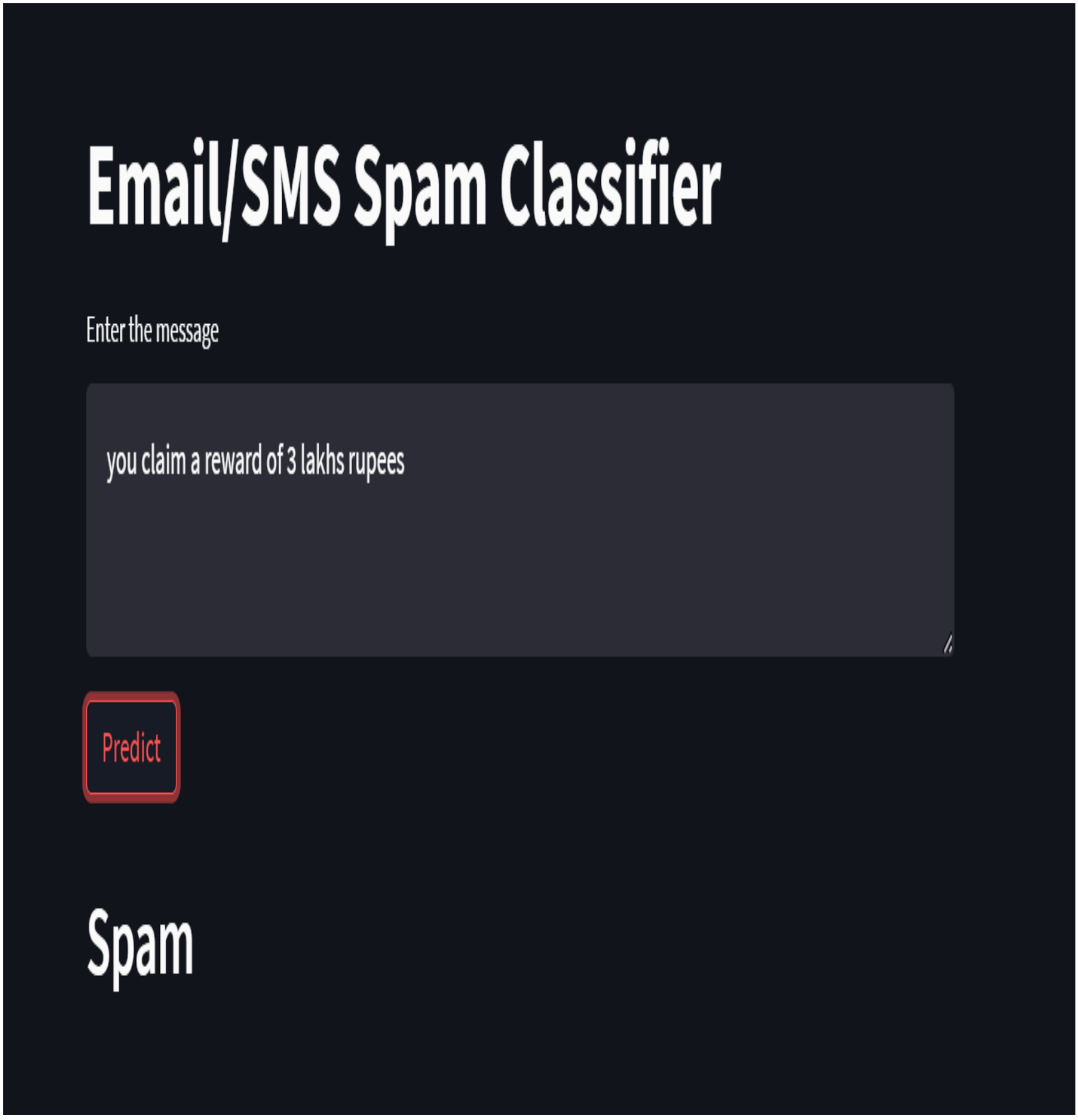
Pythor

```
Accuracy 0.9787234042553191
Precision 0.9328358208955224
```

```
import pickle
pickle.dump(tfidf,open('vectorizer.pkl','wb'))
pickle.dump(mnb,open('model.pkl','wb'))
```

Result:


Email/SMS Spam Classifier
Enter the message
hello
Predict
Not Spam


Email/SMS Spam Classifier
Enter the message
you claim a reward of 3 lakhs rupees
Predict
Spam

Conclusion

Email has been the most important medium of communication nowadays, through internet connectivity any message can be delivered to all aver the world. More than 270 billion emails are exchanged daily, about 57% of these are just spam emails. Spam emails, also known as non-self, are undesired commercial or malicious emails, which affects or hacks personal information like bank ,related to money or anything that causes destruction to single individual or a corporation or a group of people. Besides advertising,these may contain links to phishing or malware hosting websites set up to steal confidential information. Spam is a serious issue that is not just annoying to the end-users but also financially damaging and a security risk. Hence this system is designed in such a way that it detects unsolicited and unwanted emails and prevents them hence helping in reducing the spam message which would be of great benefit to individuals as well as to the company .In the future this system can be implemented by using different algorithms and also more features can be added to the existing system.