CSCE 5290: Natural Language Processing

Project Increment - 1

Group-2

Naveed Ahmed Mohammed - 11548614
Iliyas Ahmed Mohammed - 11550253
Mohammed Abdul Qayyoom Shaik - 1532186
Eswara Reddy Thimmapuram - 11506566

Project Title

Spam Message Classification

Goals and Objectives:

- **Motivation:**

  Nearly everyone now-a-days has a smart phone that at the very least has the basic feature such as messaging. Spam messages are unsolicited text messages that are forwarded to wide group of users that are typically sent for the aim of promoting products and services without their prior permission. The goal of this study is to identify whether a given message is spam or ham from the given message by using NLP techniques and machine learning algorithms.

- **Significance:**

  These days, the number of spam messages among scams has surged dramatically. These spam messages generally lure users to provide confidential and payment information by offering fraudulent and appealing deals. Using this spam detection model, we can easily identify spam messages i.e., fake messages that users are unable to recognize.

- **Objectives:**

Here, we have collected more than 1000 messages that are identified as ham & spam filled. Then, we will clean the data by eliminating punctuations and stop words using NLTK library.

We will use lemmatization i.e., normalization technique on created tokens to retrieve root words. In addition, we will apply count vectorization to eliminate words that are infrequent in the data. Furthermore, to predict the text messages we will use "Text Summarization" method to the given input sequence and then apply naïve Bayer classifier and SVM classifier from python "sklearn" package.

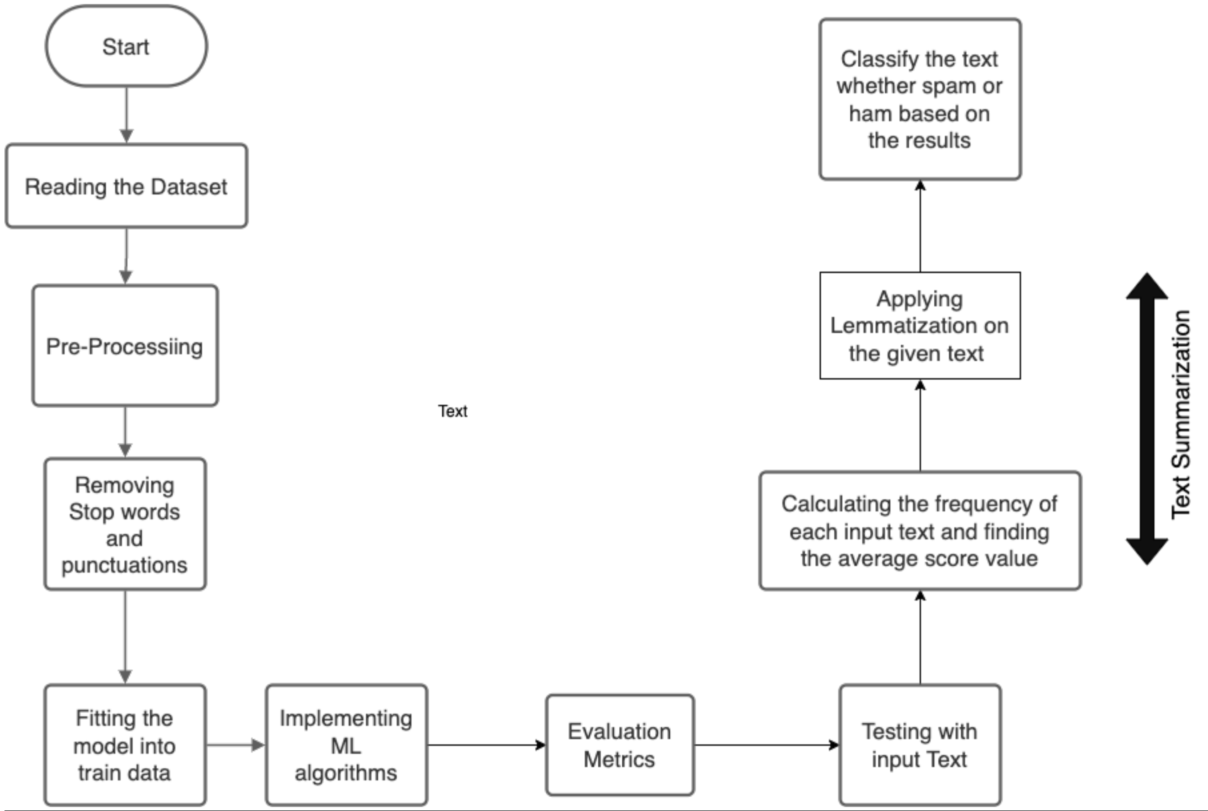
- **Features:**

The key feature of this project is to identify messages as either spam or ham. At first, the less common terms will be eliminated from the data once pre-processing and normalization techniques have been used. In order to determine if a message is spam or ham, we must train the model using the cleaned data and apply ML algorithms. As an example, the model will eliminate all the stop words and unnecessary text when we provide it with a message. Finally, after applying lemmatization to the message, the chosen algorithm will determine if the message is spam-or-ham filled.

Additional features:

- We are adding extra feature i.e., Text Summarization to this Spam Message Classification model.
  Functionality: For the input words sequence we are creating a frequency table and then tokenizing each sentence. Using the frequency method, we are finding score for each sentence and from that scores we are considering the average score of the sentences and finally generating the summary.

Workflow diagram:



References:

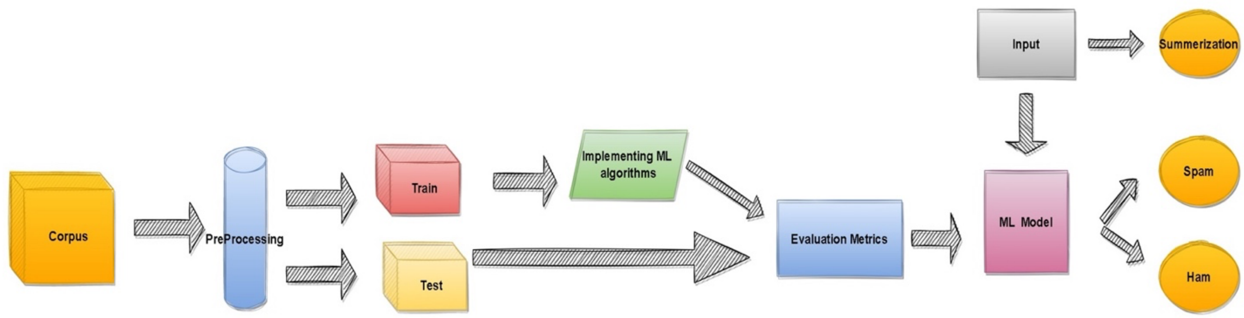
Kaggle website:
https://www.kaggle.com/datasets/team-ai/spam-text-message-classification

Journal document:
https://jusst.org/wp-content/uploads/2021/08/Classification-of-Spam-Text-using-SVM.pdf

GitHub:
https://github.com/Naveed945/Spam-Message-classifier

Increment – 1

Workflow Diagram:



• Related Work (Background):

The word ‘spam’ is defined as undesired text that is sent or received over social media platforms and messages. It is produced by spammers to divert consumers' attention from social media marketing and malware distribution, among other purposes. Spam is also present in product reviews that are posted on social networking websites. Liu & Pang (2018) estimate that between 30 and 35 percent of internet reviews are spam[1]. In a work published in 2018, "SMS Spam Filtering Using Supervised Machine Learning Algorithms," the techniques for categorizing spam messages are described. They used supervised machine learning algorithms like SVM and max entropy and performance has been accessed[2]. The classification of spam using SVM algorithm is summarized in the ‘Email Spam Classification by SVM’. In this paper, the performance of several kernel types has been assessed. The scope of this project is limited to one classification algorithm[3].

In the paper ‘A Machine Learning based Spam Detection Mechanism’ published in 2020, email spam detection was performed using Naïve Bayes algorithm including preprocessing, URL checking, tokenization and keyword checking. This paper is limited to one classification algorithm[4].

In "Email Spam Detection Using Mail Learning Techniques," which was published in 2020, different machine learning algorithms, including Naive Bayes, Support Vector Machine, Decision Tree, Neighborhood Neighbor, and Random Forest Classification, are assessed for paper spam emails. This study found that the Nave Bayes method worked well. The absence of testing the application on various data sets is a drawback of this study[5].

In "Content-Based Spam Detection in Email using Bayesian Classifier", published in 2015, the classification process is broken down into four parts in the paper: pre-processing, feature extraction, training, and classification. Its performance has been assessed, and it also explains how emails are categorized according to their content. Only one classification algorithm was covered in this study [6].

An Artificial Immune System (AIS) was created by Lutfun and Mainul[[7] for the classification of SMS. As an input spam filter, the system made advantage of a number of features. With the aid of a trained dataset that contained spam terms, phone numbers, etc., it was then utilized to categorize the text messages. The findings of this experiment demonstrated that when classifying messages as spam or not-spam, the Naive Bayesian algorithm performed superior in terms of accuracy and convergence speed. A two level stacked classifier was created by Narayan et al.[8] to distinguish between spam and legal SMS.

A selection of words whose individual probabilities are higher than a threshold are recorded in the classifier's first level. The picked words from the first level of classifier are inputted once that second level is called. They used various pairings of two-level machine learning classification algorithms, including Bayesian and SVM. Gomez et al.

[[9] examined how well Bayesian filtering techniques, which are used to prevent email spam, can be used to identify and thwart mobile spam. They preprocessed the communications using various tokenization techniques, picked out features, and evaluated their performance using several machine learning algorithms. They showed that, with the right feature extraction, Bayesian filtering approaches may be successfully used to SMS spam. In ‘Content-based SMS Spam Messages classification using Natural Language Processing and Machine Learning’ published in 2021, Sumahasan[[10] contrasted Naive Bayes with Support Vector methods using vector machines to classify SMS spam. Both models have been developed, trained, and evaluated using widely available standard datasets. The simulation's empirical findings demonstrated that the Nave Bayes-based proposed scheme outperformed the Support Vector Machine in terms of accuracy and processing speed.

For this project, we have collected some of the messages which are spam and we have understood why these messages are called spam, like they include some fraudulent offering and gifts to target the users. We have also studied about different algorithm like support vector machine(svm), random forest, naive bayer classifier and decision tree to classify the text, we have chosen the best and efficient algorithm which will help for this project.

- **Dataset:**

Dataset which we have chosen has 2 columns namely, category and message. Basically, Category has two values “ham” and “spam,” which is given for text given in message column.

This dataset contains 5573 rows filled with text and its category, which will help to better train the model.

Train set:

	A	B
1	Category	Message
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, ¬£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 ¬£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 ¬£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 that¬ís the way u feel. That¬ís 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/√∫1.20 POBOXox36504W45WQ 16+
22	ham	Is that seriously how you spell his name?

Test set:

	A
1	**Message**
2	No, I was trying it all weekend ;V
3	You know, wot people wear. T shirts, jumpers, hat, belt, is all we know. We r at Cribbs
4	Cool, what time you think you can get here?
5	Wen did you get so spiritual and deep. That's great
6	Have a safe trip to Nigeria. Wish you happiness and very soon company to share moments with
7	Hahaha..use your brain dear
8	Well keep in mind I've only got enough gas for one more round trip barring a sudden influx of cash
9	Yeh. Indians was nice. Tho it did kane me off a bit he he. We shud go out 4 a drink sometime soon. Mite hav 2 go 2 da works 4 a laugh soon. Love Pete x x
10	Yes i have. So that's why u texted. Pshew...missing you so much
11	No. I meant the calculation is the same. That <#> units at <#> . This school is really expensive. Have you started practicing your accent. Because its important. And have you decided if you are doing
12	Sorry, I'll call later
13	if you aren't here in the next <#> hours imma flip my shit
14	Anything lor. Juz both of us lor.
15	Get me out of this dump heap. My mom decided to come to lowes. BORING.
16	Ok lor... Sony ericsson salesman... I ask shuhui then she say quite gd 2 use so i considering...
17	Ard 6 like dat lor.
18	Why don't you wait 'til at least wednesday to see if you get your .
19	Huh y lei...
20	REMINDER FROM O2: To get 2.50 pounds free call credit and details of great offers pls reply 2 this text with your valid name, house no and postcode
21	This is the 2nd time we have tried 2 contact u. U have won the ¬£750 Pound prize. 2 claim is easy, call 087187272008 NOW1! Only 10p per minute. BT-national-rate.
22	Will √º b going to esplanade fr home?
23	Pity, * was in mood for that. So...any other suggestions?
24	The guy did some bitching but I acted like i'd be interested in buying something else next week and he gave it to us for free

• Detail design of Features:

This model will help to predict the message either “ham” or “spam” and to summarize the given message using the frequency table by taking the average scores of the text and then providing the summary of the message as text.

We have taken large dataset with many messages categorize into ham and spam which will help model to train better. We have taken some test set with messages which are not classified, to give input to this model and get the result in “ham” or “spam.” Finally, we will summarize the given input message and give as output with message category.

• Analysis:

After taking dataset, we can see that dataset contains noise which are with no meaning to the message. we have to remove noise with pre-processing techniques like Data cleaning which involves removal of NULL values, punctuations and stop words on dataset. Before data cleaning, for Analysis we can store the punctuation and length of the message in data frame with extra column, now we can split the column into ham and spam and calculate the mean of punctuations of spam and ham messages and mean of length of both ham and spam, We can see that the spam message has more punctuations than ham and length is also more for spam than ham.

We use stopwords and lemmatization from NLTK library for cleaning, i.e., regex to remove stopwords and if it is not present in stop words it will be lemmatize using lemmatization and after cleaning the resultant is joined to form message and is stored in data frame. The cleaned message is now used for model to train which will be more efficient.

Using TF-IDF vectorizer, we will transform the input text into meaningful representations of integers which helps the machine learning classifiers to fit into the model for better predictions. It helps in comparing the number of times a word appears in a paper to the number of documents it appears in, and can determine how original a word is.

• Implementation:

After performing data cleaning, the dataset is ready to train the model. Now, we have split the dataset into message and labels in which messages are independent values and labels are dependent values. Currently, we have divided data into test and train using sk-learn library by giving percentage to both train and test.

Moreover, we are transforming text into understandable to machine, for this we are using TF-IDF vectorizer. It is a common algorithm that converts text into a machine understandable number which is weight of words of overall document and use this to predict with machine learning algorithm.

We have used TF-IDF vectorizer, for this we have to create object of it and apply on to the data using fit_tranform, this will transform data into matrix. The matrix formed is of sentences and words. Furthermore, we are going to use pipeline concept and import it from sklearn library. This pipeline will be helpful on test data to perform all the previous steps like vectorization of the data and classifier the data in a single cell. After that we are going to implement on various classifiers like SVM, naïve bayes, etc., From that we are going to choose the classifier with more accuracy.

Once the classifier is implemented, then we can make predictions by providing various test cases and finally based on the output efficiency we can detect a good algorithm for this model.

Next, text summarization comes into picture. Firstly, we create a method for extracting the important features from the input document then collect all the keywords from the extracted features in a labelled structure. We also need to build an analyzer to detect the negative labelled features from the input document to improve accuracy. Next, to generate text summary we need to train a machine learning classifier and finally in the test phrase we generate all the relevant words and phrases and categorize them accordingly.

- **Preliminary Results:**

Visualizing the dataset in below image1. Then after we have added the punctuation length of each message in next column as shown in image2. We have cleaned the data and performed data cleaning, which involves in removal unnecessary data which is not meaningful to the message. Below is the screenshot of data cleaning, in which we have removed the null values, stopwords and punctuation as shown in below image3.

Image-1:

```
messages.head()
```

	Category	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...

Image-2:

```
messages.head()
```

	Category	Message	Length
0	ham	Go until jurong point, crazy.. Available only ...	111
1	ham	Ok lar... Joking wif u oni...	29
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...	155
3	ham	U dun say so early hor... U c already then say...	49
4	ham	Nah I don't think he goes to usf, he lives aro...	61

Image 3: before pre-processing

```
#Calculating Punctuations in each message

import string
count=0
sent=""
punct=[]
for i in range(len(messages)):
    for j in messages['Message'][i]:
        if j in string.punctuation:
            count+=1
    sent+=j
    #print(count)
    punct.append(count)
if(count!=0):
  print("Punctuations are present")
```

```
Punctuations are present
```

Image-3.1: after pre-processing we have no punctuation.

```
messages['Message'] = corpus
```

```
import string
count=0
punct=[]
for i in range(len(messages)):
    for j in messages['Message'][i]:
        if j in string.punctuation:
            count+=1
if(count==0):
  print("No punctuations")
```

```
No punctuations
```

In the below image you can see that, we have displayed the mean of punctuations respective to ‘ham’ and ‘spam.’ You can see that spam messages has more punctuations than ham.

```
[ ] spam_messages['Length'].mean()

    137.9892904953146

[ ] ham_messages['Length'].mean()

    71.44829015544042
```

We can see that spam has more words than ham

```
spam_messages['Punctuation'].mean()

5.692101740294511

[ ] ham_messages['Punctuation'].mean()

    3.939481865284974
```

spam messages has more punctuations than ham message as you can see in above reuslts.

We are categorizing the ham and spam data into labels.

```
[ ] y = messages['Category']

y.head()

0     ham
1     ham
2    spam
3     ham
4     ham
Name: Category, dtype: object
```

Once the labelling is done, we are splitting the data into train and test set. So that further we can build the model classifier.

```
[ ] X_train , X_test , y_train , y_test = train_test_split(X , y, test_size = 0.33, random_state = 42)

X_train.head()

3235                                          yup comin
945     sent score sophas secondary application school...
5319                              kothi print marandratha
5528                             effect irritation ignore
247                                         asked call ok
Name: Message, dtype: object
```

We have used TF-IDF vectorizer which transform the input text into the understandable representation of integers so that the machine learning classifier fit into the model for better predictions.

We can see in the below image.

```
X_train_tfidf_vect

array([[0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       ...,
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0]])
```

```
[ ] X_train_tfidf_vect.shape

(3733, 5772)
```

Here, we are building a model by using naïve bayes classifier.

Naive Bayer Classifier

```
[ ] from sklearn.naive_bayes import MultinomialNB
```

Below is the prediction of the test data with testing score.

```
[ ] #Predictions of the test data
    y_preds_mnb

array(['ham', 'ham', 'ham', ..., 'ham', 'ham', 'ham'], dtype='<U4')
```

```
#Training score
text_mnb.score(X_train,y_train)

0.975354942405572
```

```
[ ] #Testing score
    text_mnb.score(X_test,y_test)

0.9690048939641109
```

We are building confusion matrix and final report based on the prediction test sets.

```
[ ] print(confusion_matrix(y_test,y_preds_mnb))

[[1592    1]
 [  56  190]]
```

```
[ ] from sklearn.metrics import classification_report
```

```
print(classification_report(y_test,y_preds_mnb))

              precision    recall  f1-score   support

         ham       0.97      1.00      0.98      1593
        spam       0.99      0.77      0.87       246

    accuracy                           0.97      1839
   macro avg       0.98      0.89      0.93      1839
weighted avg       0.97      0.97      0.97      1839
```

• Project Management:

- **Work Completed:**

1) We have selected the dataset which is appropriate to the project and we have taken punctuation text into separate column and its length to understand the difference between ham and spam. To clean the noise, we have performed all the data cleaning techniques like removal of null values, stopwords and punctuations.
2) Next, we split the data in test and train in percentages which we will use in training and testing of model.
3) Then, we have used TF-IDF vectorizer which converts the string input into vectors for term frequencies and returns into machine understandable, which includes overall document to give weight of words and returned it in matrix form. Which will be used further in prediction.
4) Next, we have trained the model with naïve bayes classifier and calculated the evaluation metrics and generated the confusion matrix.

- **Tasks/Responsibilities:**

  All together we have taken some time on problem statement and researched on it and have taken appropriate dataset which better fit the requirements.

  Iliyas Ahmed (25%) – Worked on storing the dataset into data frame with name messages and given column names for it, and worked on cleaning of dataset, which involved in removal of punctuation, stop words and lemmatized the word which is not in stop words and then joined to form sentences and stored in data frame.

  Abdul Qayyoom Shaik (25%) – Handled the null values in message which are part of pre-processing. Analyzed and shown the difference in spam and ham message by taking mean of punctuation and length of the message.

  Eswara Reddy (25%) – Worked on TF-IDF vectorizer, which converted the words and sentences into machine understandable matrix for whole document and also worked on count vectorizer which is involved in count of words.

  Naveed Ahmed (25%) – Worked on Classifier and built a classifier model using naïve bayes algorithm, then I have generated the evaluated the metrics and generated the confusion matrix.

- **Work To be Completed:**

  1) Next task is to select more algorithms to classify the data and to fit into the model.
  2) Then we have to perform evaluation metrics for each algorithm to compare accuracy of each algorithm.
  3) Based on the result, we have to select algorithm with high accuracy, to get better results.
  4) Next, for input sentence we have to give to the model as input to get results and we have to apply the summarization technique to print the summary of the given input text.
  5) To summarize the input, we are using different algorithms and then performing evaluation metrics on each algorithm and then choosing the best algorithm with high accuracy.

- **Tasks/Responsibilities:**

  Iliyas Ahmed (25%) – work on the selected algorithm and start coding to accept the features from the dataset and next to develop a model. Work on text summarization, need to develop a method to extract positive and negative features from the input document.

  Eswara Reddy (25%) – After the model is developed, need to train the model by splitting the datasets into training and testing sets. And to demonstrate the count vectorizer among the testing sets.

  Naveed Ahmed (25%) – Evaluating the model using standard classifiers [Naïve bayes, SVM] by designing confusion matrix and performing metrics for the evaluation. Train a machine learning classifier to generate text summary.

  Abdul Qayyoom Shaik (25%) – After evaluating metrics, perform the predictions by providing various test cases and then compare the results and give the best suitable classifier for the summarization of the text.

References:

[1]https://www.sciencedirect.com/science/article/pii/S0957417418303749?casa_token=UD57BHqhamkAAAAA:eg0zyT9YZ5o9BR8PverPJt8QFl_m950MJYszhiLT68Q0HR1BbP9Xp96reHtwAia1axem6Oih

[2] https://ieeexplore.ieee.org/document/8442564

[3]https://www.researchgate.net/publication/220637882_A_study_of_spam_filtering_using_support_vector_machines

[4] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8802784/

[5] https://iarjset.com/wp-content/uploads/2021/06/IARJSET.2021.8632.pdf

[6]https://www.academia.edu/42943100/Improving_spam_email_detection_using_hybrid_feature_selection_and_sequential_minimal_optimisation

[7] https://www.mecs-press.org/ijitcs/ijitcs-v9-n7/IJITCS-V9-N7-5.pdf

[8]https://www.researchgate.net/publication/266654684_The_curse_of_140_characters_Evaluating_the_efficacy_of_SMS_spam_detection_on_Android

[9] https://www.ijcseonline.org/pub_paper/90-IJCSE-06523.pdf

[10]https://www.technoarete.org/common_abstract/pdf/IJERCSE/v8/i7/Ext_02649.pdf