CMPE-255 Project Fall'22 Team Project

IDENTIFICATION OF SPAM MESSAGES USING MACHINE LEARNING TECHNIQUES

Option-1: ALGORITHM TRACK

SUBMITTED BY:

Team 21

Alekhya Pasupuleti - 016622372
Nikhil konduru - 015998957
Shravani Naikoti - 016673579

Project Git Link: https://github.com/kondurunikhil/cmpe255_group

Abstract:

SMS and Emails are the widely used ways to communicate with people because of their quick transfer and low payment charges. The advancements in technology and expansion of content-based marketing have increased the use of SMS on mobile phones to advertise and attract people. This has led to an increase in spam messages. SPAM is the word commonly used to describe junk or unwanted information or messages. So, we can say that SPAM is the unsolicited SMS that is sent to the phone via text messages. The results are dangerous whether the SMS is sent through Mobile texting or using an Email. This results in leaking or exposure of personal information and privacy invasion. Although there are measures being taken to get rid of spam SMS, it has not been eliminated completely.

Through this project we are trying to identify and classify SPAM messages from legit messages using some machine learning algorithms. The classification algorithms can be used to distinguish between Spam and legit messages using the SMS spam collection dataset. We will train the machine first by providing the dataset so that it will learn from it and later make inferences on its own. It is important to detect SPAM messages to reduce the frauds that are taking place around the world.

1. INTRODUCTION:

Users are now vulnerable to cyberattacks via spam messages due to the personal and secret information they maintain on their cellphones, which includes contact lists, credit card details, photos, passwords, and much more. This compromises user privacy by allowing hackers engaged in immoral activities to acquire smartphone data without the user's knowledge. This could lead to functional and financial losses. Spam mail

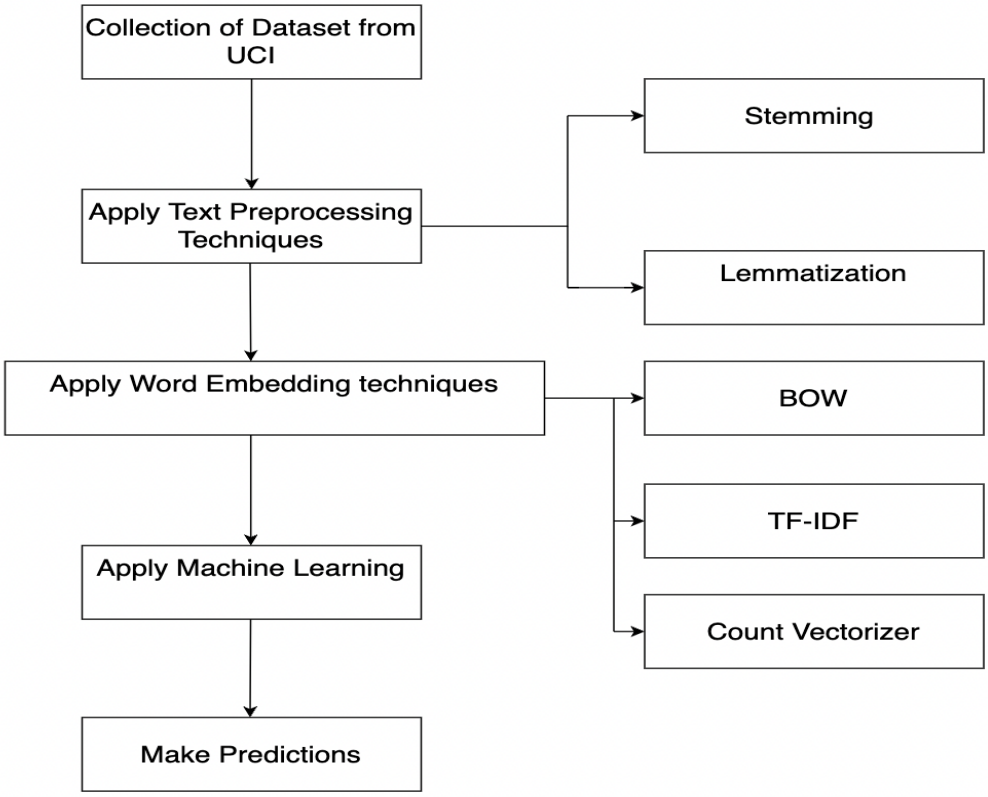
appears to be increasing, which is annoying some users and can result in serious data loss. Additionally, keyloggers and viruses can be propelled by spam messages.

These spam messages are used by spammers to advertise their services or enterprises. Due to spam mails, users may risk financial loss. Many researchers have created a set of strategies to support basic formulae, giving them an advantage over other techniques in particular. This is due to the widespread spread of this sort of problem as well as less effective security control measures.

Machine learning is a technique that enables computer systems to infer future data from historical data. Most real-world problems may now be solved using machine learning and deep learning across a range of industries, including health, security, market research, and more. On numerous datasets acquired from prior research studies, we want to compare different classification algorithms and evaluate them according to their accuracy, precision, recall, and ROC Curve. It will compare deep learning approaches to conventional machine learning techniques.

2. RESEARCH METHODOLOGY:

Below is the proposed approach for detection of SPAM messages. The project life cycle is the same for any machine learning algorithm. First, we took the raw data and tried to extract features that we already have in the data. Applied the preprocessing techniques and transformed the data into meaningful data. Later we applied machine learning models and did the feature evaluation using the models. We kept on doing this until we got the best results and after that we evaluated the models.



2.1 DATASET:

Datasets for machine learning methods are available at the UCI Machine Learning Repository. The SMS Spam Collection (also known as the corpus) is a collection of SMS-tagged messages gathered for spam message research. It includes a single batch of 5,574 English messages that have been classified as spam or ham (legal).

Each line in the files holds one message. Each line consists of two columns: the raw text plus a label (such as "ham" or "spam").

Number of Training Samples	Number of Testing Samples	Total
4180	1394	5574

Dataset link: https://archive.ics.uci.edu/ml/datasets/SMS+Spam+Collection

2.2 *PREPROCESSING:*

I. *Data cleaning*: In NLP, data cleansing is a very important step. In this, the punctuation and numeric characters are removed from the text to solely extract alphabetic characters and are reduced to lowercase. The cleaned text will then be applied to additional processing. Data cleaning Implementation link

II. *Tokenization*: Tokenization involves cutting the raw text into manageable pieces. Tokenization divides the original text into tokens, which are words and sentences. These tokens aid in context comprehension or model development for NLP. By examining the word order, the tokenization aids in deciphering the text's meaning. Tokenization Implementation link

III. *Removing Stop Words*: A language's stop words are a group of frequently used terms. Stop words in English include "a," "the," "is," "are," and others. Stop words are frequently used in text mining and natural language processing (NLP) to get rid of terms that are so frequently used that they don't contain much helpful information. Stopwords Implementation link

IV. *Stemming and Lemmatization*: Both stemming, and lemmatization aim to reduce a word's derivational forms and occasionally relate derivational forms to a basic form. The flavors of the two terms vary, though. Stemming and Lemmetization Implementation

In order to achieve this purpose, words are typically chopped off at the ends using a primitive heuristic procedure known as **stemming**, which frequently entails removing derivational affixes.

Lemmatization often refers to carrying out tasks correctly using a vocabulary and linguistic analysis of words, usually with the goal of

removing only inflectional endings and returning the lemma, or dictionary form, of a word.

The figure below shows how text looks after stemming and lemmatization.

	text	spam	length	clean	stemmed	lemmed
0	Go until jurong point, crazy.. Available only ...	0	111	[go, jurong, point, crazy, available, bugis, n...	[go, jurong, point, crazi, avail, bugi, n, gre...	[go, jurong, point, crazy, available, bugis, n...
1	Ok lar... Joking wif u oni...	0	29	[ok, lar, joking, wif, u, oni]	[ok, lar, joke, wif, u, oni]	[ok, lar, joking, wif, u, oni]
2	Free entry in 2 a wkly comp to win FA Cup fina...	1	155	[free, entry, 2, wkly, comp, win, fa, cup, fin...	[free, entri, 2, wkli, comp, win, fa, cup, fin...	[free, entry, 2, wkly, comp, win, fa, cup, fin...
3	U dun say so early hor... U c already then say...	0	49	[u, dun, say, early, hor, u, c, already, say]	[u, dun, say, earli, hor, u, c, alreadi, say]	[u, dun, say, early, hor, u, c, already, say]
4	Nah I don't think he goes to usf, he lives aro...	0	61	[nah, dont, think, goes, usf, lives, around, t...	[nah, dont, think, goe, usf, live, around, tho...	[nah, dont, think, go, usf, life, around, though]

V. *Bag of Words*: A bag of words is a textual illustration that shows where words appear in a manuscript. There are two components: a collection of well-known words. a metric for the number of well-known words.Implementation link

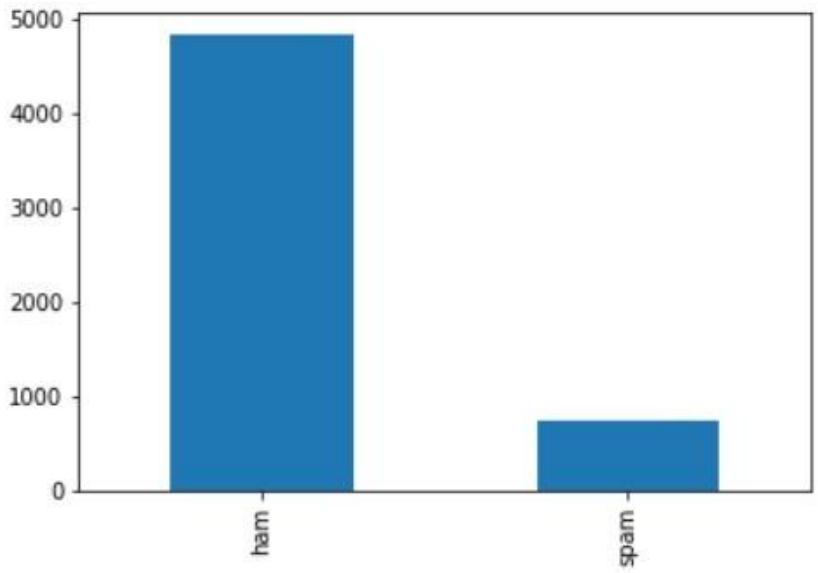
VI. *TF-IDF: Term Frequency* — Inverse Document Frequency is referred to as TF-IDF. This method counts the number of words in a collection of documents. Each word is typically given a score to indicate how important it is to the document and corpus. Information retrieval and text mining applications frequently use this method. TF-IDF Implementation link

TF-IDF = Term Frequency (TF) * Inverse Document Frequency (IDF)
Terminology: t: term (word)
d: document (set of words)
N: count of corpus
Corpus: the total document set

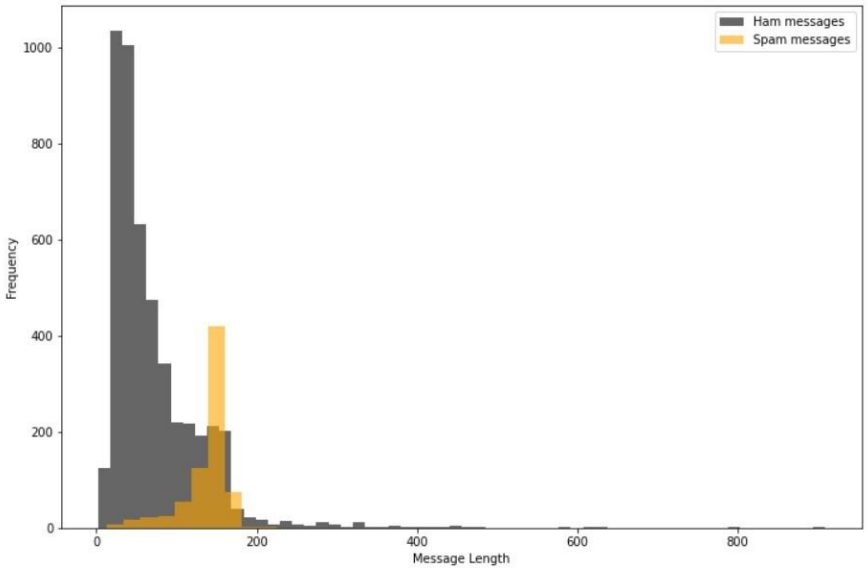
Preprocessing Contributors	Work Links
Alekhya Pasupuleti	Preprocessing_Alekhya
Alekhya Pasupuleti	Preprocessing_Alekhya
Nikhil Konduru	Preprocessing_Nikhil
Shravani Naikoti	Preprocessing_Shravani

2.3 *DATA VISUALIZATION:*

The graphic display of information and data is known as data visualization. Data visualization tools make it simple to examine and understand trends, outliers, and patterns in data by using visual elements like charts, graphs, and maps.



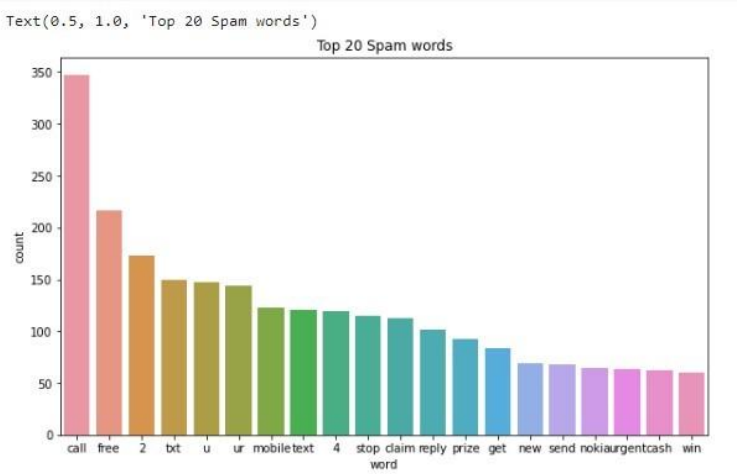
The above image shows that the number of ham messages are much more than spam messages.



From the above image we can observe that spam messages are generally longer compared to ham messages.
Spam has a length above 100 while most of the ham falls below 100.

We carried out feature engineering to visualize the data. Each message's length was computed.

The dataset's categorization uses text output in the form of the word's "spam" and "ham."



From the above image we see that popular words in spam communications include "call," "free," and misspelled or abbreviated words.

Data Visualization Contributors	Work links
Alekhya Pasupuleti	Data Visualization_Alekhya
Nikhil Konduru	Data Visualization_Nikhil
Nikhil Konduru	Data Visualization_Nikhil
Shravani Naikoti	Data Visualization_Shravani

3.Methods:

We used GridSearchCV to tune hyperparameters while training our models. It is a method for determining the best parameter values in a grid given a collection of parameters. It functions fundamentally as a cross-validation technique with the model and parameters serving as inputs. Predictions are created after obtaining the best parameter values.

I. *Multinomial Naïve Bayes*

Data analytics for email spam filtering employs the probabilistic

technique known as Naive Bayes, which is based on the Bayes Theorem. Popular in Natural Language Processing is the Bayesian learning method known as the Multinomial Naive Bayes algorithm (NLP). Using the Bayes principle, the computer makes an educated prediction about the tag of a text, such as an email or news article. It determines the likelihood of each tag for a particular sample and outputs the tag with the highest likelihood.

We have a frequency as a feature in our situation. We employ multinomial Naive Bayes in this case. It disregards the absence of the features. Therefore, multinomial naive Bayes ignores that feature if the frequency of that feature is 0, which implies that the probability of its occurrence is also 0.

II. *Random Forest*

A meta-learner made up of numerous individual trees is called a random forest. The random forest approach selects the classification with the most votes out of all the classifications for the entire data set. Each decision tree is constructed using replacement sampling from a random subset of the training dataset. In other words, certain things will show up multiple times in the sample while others won't. Each decision tree is constructed using a model based on a distinct random subset of the training dataset and a random subset of the available variables in order to discover the best way to partition the dataset at each node. Every decision tree is created to meet its largest size possible without pruning. The Random when mixed forest decision tree models result in the ensemble's final model, where each decision tree casts a vote for the result and the majority prevails.

III. *SVC*

A supervised machine learning approach called the Support Vector Machine (SVM) can be applied to classification or regression applications. However, classification issues are where it is most

frequently utilized. Each piece of data is represented as a point in an n-dimensional space, where n is the number of features you have, and each feature's value corresponds to a particular location in the SVM method. Then, classification is carried out by identifying the hyperplane that most effectively separates the two classes.

IV. *K – Nearest Neighbours*

An efficient and straightforward classification algorithm is the KNN algorithm. The key challenge with this algorithm is predicting the initial value of k. Performance of the algorithm depends on the k value picked. The labels for the test data are not predicted using the trained model in a separate stage after the training stage in KNN. Instead, the K nearest training data items is picked, the most common class among them is assigned to the test data item, and the features of each test data item are compared in real time with the features of each training data item.

Contributors	Work links
Alekhya Pasupuleti	KNN_Alekhya.ipynb
Alekhya Pasupuleti	Logistic_Regression_Alekhya.ipynb
Alekhya Pasupuleti	MNB_Stemming_Lemmatization_Alekhya.ipynb
Alekhya Pasupuleti	Multinomial_Naive_Bayes_Alekhya.ipynb
Alekhya Pasupuleti	RF_Alekhya.ipynb
Alekhya Pasupuleti	SVC_Alekhya.ipynb
Alekhya Pasupuleti	sms_spam_final.ipynb
Nikhil Konduru	Confusion_Matrix_Nikhil.ipynb

Nikhil Konduru	KNN_Nikhil.ipynb
Nikhil Konduru	Model_Evaluation_Nikhil.ipynb
Nikhil Konduru	Multinomial_Naive_Bayes_Nikhil.ipynb
Nikhil Konduru	ROC_Curves_Nikhil.ipynb
Nikhil Konduru	Random_Forest_Nikhil.ipynb
Nikhil Konduru	SMS_Spam_Classifier_Nikhil_Final.ipynb
Nikhil Konduru	SVM_Nikhil.ipynb
Shravani Naikoti	KNN_Final_Shravani_Naikoti.ipynb
Shravani Naikoti	Random_Forest_FInal_Updated_Shravani.ipynb
Shravani Naikoti	Random_Forest_Final_Shravani.ipynb
Shravani Naikoti	SVC_FINAL_Shravani_Naikoti.ipynb
Shravani Naikoti	Trained_Final_Models_Shravani.ipynb

4.COMPARISON

4.1 Performance Metrics

4.1.1 Accuracy:

When the quantity of samples in the various classes differs significantly (unbalanced target), accuracy is not a good indicator of a classifier's genuine performance because it will lead to false conclusions.

$$Accuracy = \frac{\text{Number of correct predictions}}{\text{Total of all cases to be predicted}}$$

4.1.2 Precision:

The capability of a classification model to identify only relevant data points. According to mathematics, precision is calculated by dividing the total number of true positives by the total number of true positives + false positives.

$$precision = \frac{true\ positives}{true\ positives + false\ positives}$$

4.1.3 Recall:

The capacity of a model to locate all pertinent instances in a data set. Recall is calculated mathematically as the sum of the true positives and false negatives divided by the total number of true positives.

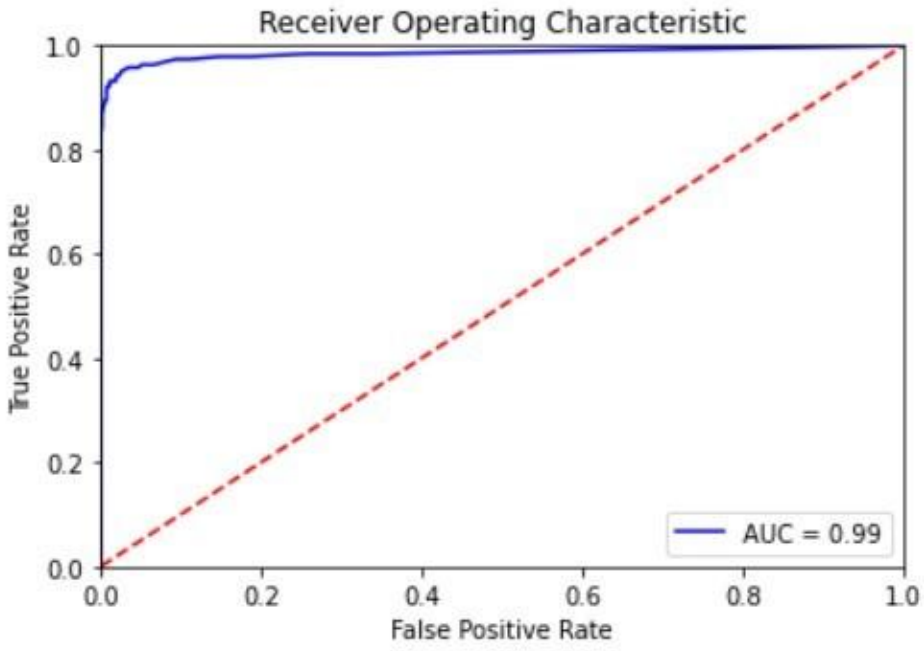
$$recall = \frac{true\ positives}{true\ positives\ + false\ negatives}$$

4.1.4 F1-score:

In the following equation, where both metrics are taken into consideration, the F1 score is the harmonic mean of precision and recall.

4.1.5 AUC-ROC:

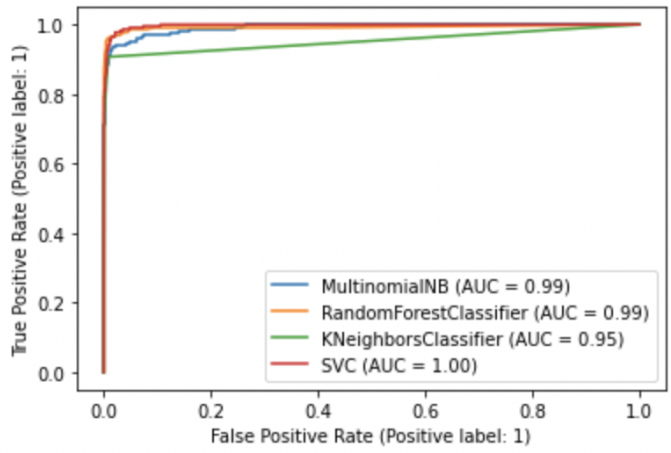
A performance statistic for classification issues at different threshold levels is the AUC-ROC curve. AUC is a measurement of separability, whereas ROC is a probability curve. It shows how effectively the model can differentiate across classes.



4.2 Comparison

The Random Forest Classifier fared the best out of all the models we used. As you can see from the table below, Random Forest has a very high F1 score.

	Accuracy	Precision	Recall	F1 score	ROC AUC
Random Forest	**0.97**	**1.0**	**0.97**	**0.99**	**0.99**
SVC	**0.97**	**1.0**	**0.97**	**0.98**	**1.0**
KNN	**0.89**	**1.0**	**0.90**	**0.94**	**0.95**
Naive Bayes	**0.96**	**1.0**	**0.96**	**0.98**	**0.99**



Final Project Code link: sms_spam_final.ipynb

5. CONCLUSION

In this article, machine learning-based spam filtering methods are covered. We utilized a UCI dataset for our experiments. The only dataset utilized to test our model was this one.

We employed the Bag of Words, Count vectorizer, and TF-IDF word embedding techniques. The TF-IDF with Random Forest classification approach outperforms the other algorithms in the trial in terms of accuracy score.

Due to the imbalanced nature of the dataset, it is not sufficient to evaluate the performance solely on accuracy; the precision, recall, ROC score, and f1-score of the algorithms must also be considered.

The Random Forest algorithm continues to perform well in additional testing, with precision of 0.97, ROC score of 0.99, and f1-score of 0.99. In order to balance the dataset, undersampling was used. Even after undersampling, the Random Forest algorithm was still able to deliver good results. The performances and results of various algorithms will vary depending on the criteria used. Classifier performance and training data

may both be enhanced by including additional variables. In the future, we'll test our model using a range of datasets.

6. Teamwork

Each team member contributed equally, and all deliverables were completed with adequate coordination and due consideration for each person's ideas during project implementation.

7. References

1. D. K. Renuka, T. Hamsapriya, M. R. Chakkaravarthi and P. L. Surya, "Spam Classification Based on Supervised Learning Using Machine Learning Techniques," 2011 International Conference on Process Automation, Control and Computing, 2011, pp. 1-7.
2. Shrawan Kumar Trivedi, “A Study of Machine Learning Classifiers for Spam Detection”, 2016 4th International Symposium on Computational and Business Intelligence,2016
3. Naeem Ahmed, Rashid Amin, Hamza Aldabbas, Deepika Koundal, Bader Alouffi, and Tariq Shah “Machine Learning Techniques for Spam Detection in Email and IoT Platforms: Analysis and Research Challenges”, 3 February 2022 Hindawi Security and Communication Networks Volume, 2022
4. Mehul Gupta, Aditya Bakliwal, Shubhangi Agarwal & Pulkit Mehndiratta , “A Comparative Study of Spam SMS Detection using Machine Learning Classifiers”, Proceedings of 2018 Eleventh International Conference on Contemporary Computing (IC3), 2018
5. https://towardsdatascience.com/your-guide-to-natural-language-processing-nlp-48ea2511f6e1
6. https://towardsdatascience.com/email-spam-detection-1-2-b0e06a5c0472