International Journal of Recent Advances in Multidisciplinary Topics
Volume 3, Issue 5, May 2022
https://www.ijramt.com | ISSN (Online): 2582-7839

Spam Detection Using Artificial Intelligence

Chetan Malik[1*], Deepak Pal[2], Vishu Verma[3]
[1,2,3]*Department of Computer Science and Engineering, Meerut Institute of Engineering and Technology, Meerut, India*

***Abstract*: Now-a-days, the most majority of people depend on what is available email or messages sent by a stranger. Possibly anyone can leave an email or message provide gold the opportunity for spam senders to write a spam message about us different interests. Spam fills in the inbox with a number of funny things emails. Slow down our internet speed. Theft useful information such as our details on our contact list. Identifying these people who post spam and spam content can be a hot topic for research and strenuous activities. Email Spam is functionality of mass mailings. From the cost of Spam is heavily censored by the recipient, it is a successful post proper advertising. Spam email is a form of commercial advertising economically viable because email can be costly effective sender method. With this proposed model I some message may be declared spam or not use Bayes' theorem and Naive Bayes 'Classifier and IP addresses of sender is usually found.**

***Keywords*: Electronic emails, Email spam, Spam detection, Machine Learning, Support Vector Machine, Naïve Bayes classifier.**

1. Introduction

In recent years, the Internet has become an important part of our life. With more internet usage, the email user numbers are the same they grow day by day. This increasing use of email has increased created problems caused by unsolicited bulk email messages commonly called Spam. Email is now private of the best ways to advertise because of spam emails there is produced. Spam emails are created by the recipient I do not wish to accept. a large number of similar messages sent to a few email recipients. Spam usually appears as the result of providing our email address to an unauthorized location or unreliable website. There are too many Spam results Fills in the Inbox with a number of funny emails. Because our internet speed on a large scale. Be helpful information as our details in the contact list. Changes your search results on any computer program. Spam is a huge waste everyone and can get frustrated quickly if you get great value. Seeing these spam senders and spam content is hard work. However, a large number of studies have been conducted, however to date the prescribed methods do not separate spam surveys, either none of them show the benefits of each deleted name feature. In addition to network coverage communication and wasting a lot of memory space, spam messages are used to attack something. Spam emails, too known as non-personal, unsolicited or malicious commercial emails, sent to contact one person or a company or a crowd of people. Apart from advertising, these may contain links to sensitive phishing scams or malicious websites. found to steal confidential information. to solve this problem different spam filtering methods are used. The spam filtering techniques often protect our mailbox in spam mails.

2. Literature Survey

In paper [1], the authors highlight a few features contained in the email subject that will be used for identification and classify spam messages correctly. These features are selected based on their performance in detecting spam messages. This paper also covers individual features contains email from Yahoo, Gmail and Hotmail so you have regular spam messages the find method may be suggested for all major email's providers. In Paper [2], a new method based on the strategy how often words are repeated. The key sentences, those with keywords, of incoming emails should be marked and after that the grammatical roles of complete words in a sentence need to be cut, finally they will be added to the vector to take similarities between received emails. K-Mean algorithm used to classify received email. Vector's determination is method used to determine the category of emails. In paper [3], the authors describe the cyber-attacks. But predators and attackers often use email services to send false types of user-directed messages.

They may lose their money and their reputation in the community. These result in to obtain personal information such as a credit card number, passwords and some confidential data. In this paper, the authors use Bayesian Categories. Consider each single word in the post. It is always getting used to new types of spam. In paper [4], the proposed system attempts to operate the machine learning strategies to find repetitive keyword pattern classified as spam. The program also suggests classification of emails based on various other parameters contained in their structure such as Cc / Bcc, domain and header. Each parameter will be considered as a feature

what to use it in a machine learning algorithm. The machine the learning model will be a pre-trained model with feedback how to distinguish between right output and obscure output. This method provides an alternative property with which spam filters can be used. This paper also considers the body of the email with the most commonly used keywords and punctuation marks. In paper [5], the authors investigated the use of string the same algorithms for receiving spam email. Especially this work evaluates and compares the effectiveness of the six well-being.

Communication experts estimate that 40% of social media

*Corresponding author: chetan.malik.cs.2018@miet.ac.in accounts used for spam [8]. Spam senders use popular social networking tools to direct certain sections, review pages, or fan pages to post hidden links in text to pornographic sites or other product sites designed to sell something from fraudulent accounts. Harmful emails are sent to the same type of people or organizations sharing the most common photos. By researching these excellent images, one can improve the discovery of these types of emails. By using artificial intelligence (AI) [9], we can split emails into spam and non-spam emails. This solution is possible by using the feature extracted from the message headers, title, and body. After extracting this data based on its nature, we can combine it with spam or ham. Today, study-based categories [10] are often used for spam detection. In a study-based classification, the discovery process assumes that spam emails have a specific set of characteristics that distinguish them from official emails [11]. Many factors increase the complexity of the spam detection process in learning-based models. These factors include spam submission, visual erosion, language problems, high processing, and text delays.

3. Methodology

The process for implementing a project is as follows given below in the form of a flowchart. Also, details of document processing are listed below.

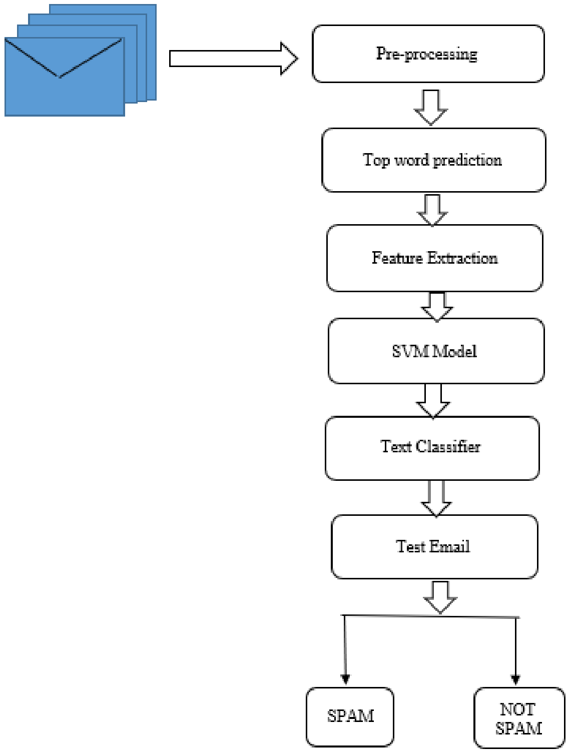


Fig. 1. Processing flowchart

A. *Pre-processing*

The pre-processing step is used to remove the noises from the email which are irrelevant and need not be present. The pre-processing step includes:

- Removal of Numbers
- Removal of Special Symbol
- Removal of URLS
- Stripping HTML
- Word Stemming

B. *Feature Extraction*

To exclude important and relevant features from email body Feature Domain is used. Feature converts email into a 2D vector with features number. These features are mapped to a list of names.

The Vector feature of the email can be defined as follows:

x = [0.123 0.523 0.428 0.902. . . . . 0.014 0.890]

Feature Vector is defined by calculating TF-IDF values. TF IDF stands for Term Frequency - Inverse Document Usually. It is calculated using the following formula.

tf-idf (t, d) = (n (t, d) / n (d)) * log (N / n (t))

There, n (t, d) = Number of times a word (term) appears in email (document) d
n(d) = Total Number of words in the email.
N = Total Number of emails (documents) in the training set.
n(t) = Total Number of emails those contain the word t.

C. *SVM Model*

Support Vector Machine is used for separation as well and with regression problems where the database is used train SVM to separate any new data it receives. It is a learning machine algorithm that works detecting a hyperplane that separates the database separately classes. SVM increases the distance between variations classes due to the presence of multiple line hyperplanes called margin enlargement. SVM has confirmed that it is one of the most economically viable alternatives kernel techniques. The success of SVM is largely due to its high level of normalization. good rent a straight kernel within the SVM can be considered as an affiliate embedding of the input location into a higher dimensional enter the location wherever the separation is made on time not explicitly exploiting this feature instead of spam email used for training purposes. The training database contains spam content and separator are trained through use. After training, the separator is ready to separate spam emails.

D. *Naïve Bayes Classifier (NB)*

The Naïve Bayes classifier [12] is based on the Bayes theorem. It assumes that the predictors are independent, which means that knowing the value of one attribute impacts any other attribute’s value. Naïve Bayes classifiers are easy to build because they do not require any iterative process and they perform very efficiently on large datasets with a handsome level of accuracy. Despite its simplicity, Naïve Bayes is known to have often outperformed other classification methods in various problems. [13] present research on email spam filtering and perform the analysis using a machine learning algorithm Naïve Bayes. They used two datasets evaluated on the value of accuracy, F-measure, precision, and recall. As we know, Naïve Bayes uses probability for classification, and the probability is counting the frequency and combination of values in a dataset. This research uses three steps for the filtration of emails, i.e., preprocessing, feature selection, and, at last, it implements the features by using the Naïve Bayes classifier. The preprocessing step removes all conjunction words, articles, and stop words from the email body. Then, they used the WEKA tool [14] and made two datasets called spam data and spam base dataset. The average accuracy was 89.59% using two datasets, while the spam data got 91.13% accuracy.

E. Test Classifier

To check the accuracy of the classification, the separator is checked and numerous training data. Test data set for samples from the database can be used for training.Here 30% data is used for testing purposes. And data I randomly selected in the database. Up to 94% accuracy in separating emails is achieved with the proposed solution.

1) Test Email

After the completion of the training phase, a new email sample states provided as a classification input to split email. The separator produces output in the range of 0 or 1, 1 means that it is spam and 0 means it is not spam.

F. Documentation Processing

1) Tokenization

Tokenization is the process of breaking the text stream at the top in words, phrases, symbols, or other important parts tokens are called. The list of tokens becomes a continuous entry processing such as analyzing or digging a text. Typically, token-making takes place at the word level. However, it is sometimes difficult to define what the word "word" means. A tokenizer relies on simple heuristics, for example:

- All connected letters of the alphabet are part of single character; as well as numbers.
- Tokens are separated by white space letters, such as place or line break, or with punctuation marks

In languages such as English (and many editing languages) when words are separated by a white space, this method is easy. However, tokenization is difficult in such languages as the Chinese have no word boundaries. It's easy making tokens divided into white areas also brings difficulties where word collections such as NY should be treated together mark. By creating complex heuristics, asking a Table of common assignments, or inserting tokens in a language model identifying collections in recent times Steps to consider are some of the ways to deal with this problem.

For example:

Input: “Practice perfection”

- Output: Tokens
- Exercise
- Do
- a man
- complete

An example of a sequence of letters is called a Token.

2) Stemming:

Stemming is the process of reducing a word to its word stem that affixes to suffixes and prefixes or the roots of words known as a lemma. When a new word is found, it can present new research opportunities. Often, the best results can be attained by using the basic morphological form of the word: the lemma. To find the lemma, stemming is performed by an individual or an algorithm, which may be used by an AI system. Stemming uses several approaches to reduce a word to its base from whatever inflected form is encountered. To develop a stemming algorithm, lemma can be used. Some simple algorithms will simply strip recognized prefixes and suffixes. However, these simple algorithms are prone to error. For example, an error can reduce words like laziness

instead of lazy. Such algorithms may also have difficulty with terms whose inflectional forms don't perfectly mirror the lemma such as with saw and see.

3) Lemmatization

Lemmatization in languages is the process of integration together the different types of word have been changed to be the same it is usually analyzed as a single object. Algorithmic process for determining lemma for The name given is Lemmatization, in Linguistics. Since the method may involve complex tasks such as comprehension context and determining part of the word expression at a time sentence (which requires, for example, the knowledge of grammar) is often a difficult task to perform a lemmatizer for a replacement language.

4) Removal of Stop Word

Sometimes, the most common word is possible appears to have little or no benefit in assisting the selected documents The customer's user requirement is removed from the dictionary. These words are called words stop so the process is called cessation removal. A general strategy for determining the filter stop list goals by collection usually then make the priority often in terms of terms, such as a list of stops, its members discarded at the time of identification. Some samples of the set name are: a, an, the, and, are, as, that, be, for, from, to, to, is, of, of, to, to, continue, was, he was there, he was going to do, with, etc.

4. Results

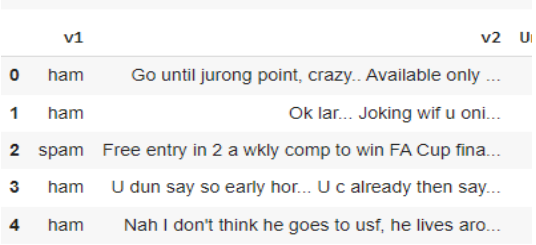
A. Dataset after Preprocessing

Dataset after having performed preprocessing steps such as:

1) Removal of numbers
2) Removal of html tags
3) Removal of punctuation marks
4) Removal of stop words
5) Stemming and Lemmatization
6) Converting all data to lower case

Preprocessed dataset is shown in fig. 2.

Figure represents a database framework, containing 2 columns. The first column v2 contains an email text after pre-processing while the second column represents the appropriate label.



	v1	v2	U
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...	

Fig. 2. Preprocessed dataset

B. *Visualizing the Data*

Studying the dataset and visualizing as per the requirement.

1) *Spam vs. Ham Value Count*

Total number of spam emails and total number of non-spam or ham emails in the entire dataset are shown in the bar graph Fig. 3. below**.**

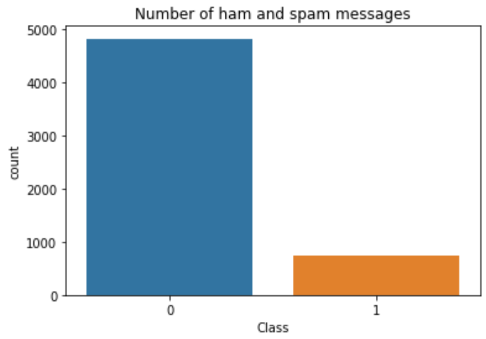
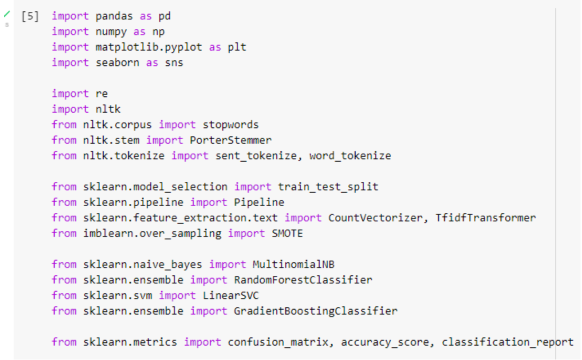


Fig. 3. Spam vs. Ham value count

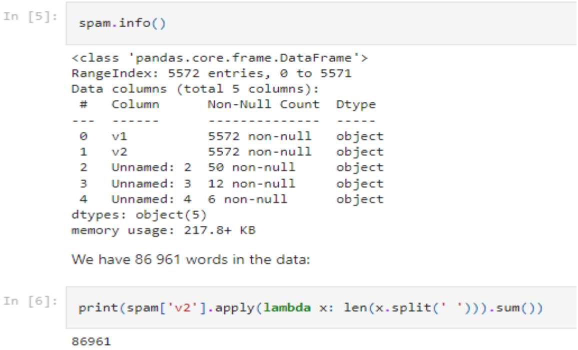
With the help of NLTK (Natural Language Toolkit) for to process text, Using Matplotlib you can edit graphs, histogram and bar layout and all that stuff, Word Cloud is used to present text data and pandas for data access cheating and analysis, NumPy is doing it mathematical and scientific performance. Packages used in the proposed model are displayed



```
[5] import pandas as pd
    import numpy as np
    import matplotlib.pyplot as plt
    import seaborn as sns

    import re
    import nltk
    from nltk.corpus import stopwords
    from nltk.stem import PorterStemmer
    from nltk.tokenize import sent_tokenize, word_tokenize

    from sklearn.model_selection import train_test_split
    from sklearn.pipeline import Pipeline
    from sklearn.feature_extraction.text import CountVectorizer, TfidfTransformer
    from imblearn.over_sampling import SMOTE

    from sklearn.naive_bayes import MultinomialNB
    from sklearn.ensemble import RandomForestClassifier
    from sklearn.svm import LinearSVC
    from sklearn.ensemble import GradientBoostingClassifier

    from sklearn.metrics import confusion_matrix, accuracy_score, classification_report
```

Fig. 4. Train dataset

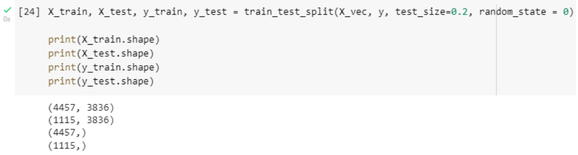
1. Split the data into training and testing sets as shown below. Some percentage the data set is used as train dataset and the rest as a test dataset.



```
In [5]:  spam.info()

         <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

         We have 86 961 words in the data:

In [6]:  print(spam['v2'].apply(lambda x: len(x.split(' '))).sum())

         86961
```

Fig. 5

2. Reset the train with the test indicator as shown in the following column:



```
[24] X_train, X_test, y_train, y_test = train_test_split(X_vec, y, test_size=0.2, random_state = 0)

     print(X_train.shape)
     print(X_test.shape)
     print(y_train.shape)
     print(y_test.shape)

     (4457, 3836)
     (1115, 3836)
     (4457,)
     (1115,)
```

Fig. 6. Reset train and test index

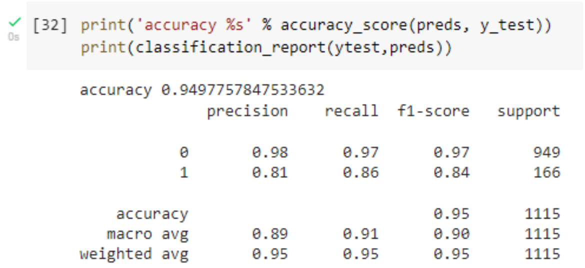
3. Whenever there is a message, we must first consider It input messages. We need to change all inputs small capital letters.
4. Then divide the text into smaller pieces and retrieve it punctuation marks. So, the Tokenization process is used remove punctuation marks and split messages.
5. The Porter Stemming Algorithm is used for deterrence. Stemming is the process of reducing words to their origin voice.
6. We need to find opportunities for word spam once ham messages.
7. Tf -idf (term term-inverse document frequency) contains to be reckoned with.
   - TF: Term Frequency, which measures how many times the name comes from the document.
   - TF (t) = (Number of times t appeared in the document) / (Total terms in the document).
   - IDF: Inverse Document Frequency, measuring I the significance of this word.
   - IDF (t) = loge (Total text/text with the word t in it).
8. See how well the model is done explores the Naïve Bayes Classifier and shows up.

5. Discussions

If we receive a message in the inbox, that message will be sent to the database as shown below. This message will be detected as spam or not.

The message sent will be received as spam or unused. The theory of Bayes and the Naive Bayes' Classifier follows it all the steps mentioned above and finding opportunities to spam words and ham messages to see if it is spam or not. The statistics below show a message received as

If "Thanx" is a message sent from the Inbox to the dataset then uses the Bayes' and the Naive Bayes' theorem 'Separator, the message above is received as Ham as shown below.

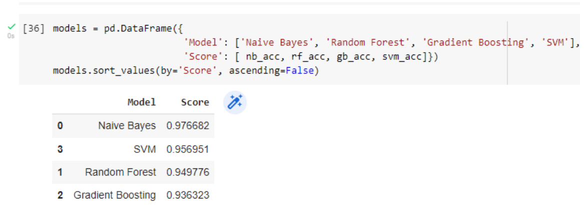


```
[32] print('accuracy %s' % accuracy_score(preds, y_test))
     print(classification_report(ytest,preds))

     accuracy 0.9497757847533632
                   precision    recall  f1-score   support

                0       0.98      0.97      0.97       949
                1       0.81      0.86      0.84       166

         accuracy                           0.95      1115
        macro avg       0.89      0.91      0.90      1115
     weighted avg       0.95      0.95      0.95      1115
```

Fig. 7. Ham message

Best model:

We tested four different models and now, we check which one is the best:



```
[36] models = pd.DataFrame({
                    'Model': ['Naive Bayes', 'Random Forest', 'Gradient Boosting', 'SVM'],
                    'Score': [ nb_acc, rf_acc, gb_acc, svm_acc]})
     models.sort_values(by='Score', ascending=False)

                   Model     Score
     0       Naive Bayes  0.976682
     3               SVM  0.956951
     1     Random Forest  0.949776
     2 Gradient Boosting  0.936323
```

Fig. 8.

6. Conclusion

Email has become a very important means communication today; through any internet connection the message can be sent to all parts of the world. More than 270 billion emails are changed daily, about 57% of these just spam emails. Spam emails, also known as non-spam, unwanted or malicious marketing emails, touch or hacks personal information such as banking, money-related or anything that causes damage to one person or a company or group of people. Apart from advertising, these may contain links to sensitive phishing scams or malicious websites.is set to steal private information. Spam is dangerous an issue that not only annoys end users but also financially harmful and security risks. So, this program is designed to detect unsolicited one's emails are unwanted and blocking them so help reducing spam message which can be very beneficial to individuals and to the company. In the future this system can be implemented using different algorithms at once and additional features can be added to the existing system.

References

[1] Shukor Bin Abd Razak, Ahmad Fahrulrazie Bin Mohamad "Identification of Spam Email Based on Information from Email Header" 13th International Conference on Intelligent Systems Design and Applications (ISDA), 2013.

[2] Mohammed Reza Parsei, Mohammed Salehi "E-Mail Spam Detection Based on Part of Speech Tagging" 2 nd International Conference on Knowledge Based Engineering and Innovation (KBEI), 2015.

[3] Sunil B. Rathod, Tareek M. Pattewar "Content Based Spam Detection in Email using Bayesian Classifier", IEEE ICCSP 2015 conference.

[4] Aakash Atul Alurkar, Sourabh Bharat Ranade, Shreeya Vijay Joshi, Siddhesh Sanjay Ranade, Piyush A. Sonewa, Parikshit N. Mahalle, Arvind V. Deshpande "A Proposed Data Science Approach for Email Spam Classification using Machine Learning Techniques", 2017.

[5] Kriti Agarwal, Tarun Kumar "Email Spam Detection using integrated approach of Naïve Bayes and Particle Swarm Optimization", Proceedings of the Second International Conference on Intelligent Computing and Control Systems (ICICCS), 2018.

[6] Cihan Varol, Hezha M. Tareq Abdulhadi, "Comparison of String-Matching Algorithms on Spam Email Detection", International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism, Dec. 2018.

[7] Duan, Lixin, Dong Xu, and Ivor Wai-Hung Tsang. "Domain adaptation from multiple sources: A domain dependent regularization approach." IEEE Transactions on Neural Networks and Learning Systems 23.3, 2012.

[8] K. S. Adewole, N. B. Anuar, A. Kamsin, K. D. Varathan, and S. A. Razak, "Malicious accounts: dark of the social networks," Journal of Network and Computer Applications, vol. 79, pp. 41–67, 2017.

[9] A. Barushka and P. Hájek, "Spam filtering using regularized neural networks with rectified linear units," in Proceedings of the Conference of the Italian Association for Artificial Intelligence, Springer, Berlin, Germany, November 2016.

[10] F. Jamil, H. K. Kahng, S. Kim, and D. H. Kim, "Towards secure fitness framework based on IoT-enabled blockchain network integrated with machine learning algorithms," Sensors, vol. 21, no. 5, p. 1640, 2021.

[11] M. H. Arif, J. Li, M. Iqbal, and K. Liu, "Sentiment analysis and spam detection in short informal text using learning classifier systems," Soft Computing, vol. 22, no. 21, pp. 7281–7291, 2018.

[12] I. Rish, "An empirical study of the naive Bayes classifier," in IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence, University of British Columbia, Computer Science Department, Vancouver, Canada, 2001.

[13] N. F. Rusland, N. Wahid, S. Kasim, and H. Hafit, "Analysis of Naïve Bayes algorithm for email spam filtering across multiple datasets," in Proceedings of the IOP Conference Series: Materials Science and Engineering, IOP Publishing, Busan, Republic of Korea, 2017.

[14] A. K. Sharma and S. Sahni, "A comparative study of classification algorithms for spam email data analysis," International Journal on Computer Science and Engineering, vol. 3, no. 5, pp. 1890–1895, 2011.