HBRP PUBLICATION

Journal of Advanced Research in Artificial Intelligence & It's Applications
Volume 2 Issue 1
e-ISSN: 3048-6440

Artificial Intelligence-Based Email Spam Filtering

Low Choon Keat[1]*, Tan Xuan Ying[2]
[1]Faculty, [2]Reasercher of Computing and Information Technology (FOCS), Tunku Abdul Rahman University of Management and Technology (TAR UMT), Jalan Genting Kelang, Setapak, Kuala Lumpur, Malaysia

****Corresponding Author***
Email Id: *lowck@tarc.edu.my*

ABSTRACT

Email spamming has become a big issue in spreading unsolicited emails. This project focuses on tackling the prevalent issue of email spam through the implementation of machine learning techniques, particularly emphasizing spam filtering using artificial intelligence (AI). The goal is to develop an AI-powered application for efficient identification and filtration of spam emails. Key functionalities include email content preprocessing, feature extraction using techniques like Count Vectorization and TF-IDF Vectorization, and deploying machine learning models such as Support Vector Machines, Naive Bayes, and K-Nearest Neighbour classifiers. The results demonstrate an impressive 98.65% accuracy in recognizing spam emails. The conclusion highlights the significance of AI in spam filtering for accurate and reliable outcomes, addressing the persistent problem of email spam. This research contributes to advancing spam filtering techniques, serving as a valuable reference for future email security research and development.

Keywords*:-Spam Filtering, AI, Naive Bayes, KNN, SVM, TFIDF, BoW*

INTRODUCTION

The 21st century has seen a clear dependence on technology for communicating across long distances, particularly through email. This method of communication is widely used, with 4.26 billion users and an expected 333 billion emails sent and received in 2022. Although highly efficient, the increasing number of emails has resulted in a significant rise in spam, which presents security threats due to the presence of unsolicited messages that may contain harmful malware.

The act of digital spamming can give rise to several complications and concerns for both the recipient and the overall web security. Spamming, as an illustration, led to the utilization of server resources such as time and storage space. This, in turn, occupied the bandwidth, resulting in a decrease in speed and potentially impacting the search results for wanted emails that were a mix of spam and valid messages [1] . During extensive research on web security concerns, it has been observed that hackers have employed spamming as a method to obtain information.

Cybercriminals send unsolicited emails to recipients and include harmful payloads through attachments and dangerous links in order to get valuable information from the recipient. Upon the recipient’s activation of the connection, the payload will be initiated and execute its intended operations. Currently, scammers are utilizing spamming as a means to unlawfully get personal information from receivers, jeopardizing their privacy and security [2].

To mitigate the annoyance caused by spam emails, the standard approach is to implement email spam filtering technology. It has the capability to recognize unsolicited and trash emails and mark those spam emails to prevent them from appearing in a user's inbox. Various strategies are employed in email spam filtering, including content-based filtering and list-based filtering [3].

Given the increasing prevalence of Artificial Intelligence (AI) technology, numerous security experts are investigating how AI might be utilized to improve and fortify web security in order to address security concerns. AI has recently become a novel approach in email spam filtering, aiming to enhance the effectiveness of spam detection [4] . Natural Language Processing (NLP) is an area of Artificial Intelligence (AI) focused on enabling computer programs to comprehend and interpret human language, sometimes referred to as natural language. Using natural language processing (NLP), the computer can comprehend and acquire knowledge from provided emails in order to identify and categorize any spam emails based on their content.

The aim of this research project is to focus on the development of an email spam filtering application that utilizes Artificial Intelligence (AI) to improve the accuracy of detecting and filtering spam emails. This research will specifically concentrate on Natural Language Processing (NLP), a field that empowers machines to comprehend human language as written in emails. The objective is to develop algorithms that can effectively recognize and filter spam emails. To enhance the accuracy and precision of spam email filtering methods compared to other email spam filters.

RELATED WORKS

In this section, several spam filtering models have been discussed by many re searchers.Listed several related below:

A researcher from [5] has proposed Naive Bayes algorithms with standard BoW approach, to convert text from string to numerical value and achieved a high accuracy of 97.5%. Despite the dataset the researcher working on is SMS messages labeled ham spam, containing 5574 entries and 749 spam messages in it.

Researchers from [6] carried out different machine learning algorithms to do comparison with. They proposed Naive Bayes, Support Vector Machine, Decision Tree and Random Forest to test the optimal result they can get. They also tested the algorithms with two different dataset and the optimal result was Random Forest across all other algorithms which got the highest accuracy around 97.6%.

A method proposed by [7] working on improving Support Vector Machine algorithm through tuning the hyperparameter to achieve higher results on the popular spam dataset which is Spam Base with total entries of 4601. The accuracy achieved was 94.06% which was outperforming the other researchers who worked on the Spam Base as well. [8] proposed KNN and Naive Bayes machine learning algorithms to test on Enron dataset with 5572 entries of message with 747 of spam messages in it. The result generated showed KNN has higher accuracy than Naive Bayes however Naive Bayes performed better in ham message classification than KNN.

***Table 1:**-Summary of Related Works*

Authors	Dataset	Methods	Feature Extraction	Result
Chavez (2020)	SMS Spam Collection Data Set	Naive Bayes	BoW	Accuracy 97.5%
Junnarkar et al. (2021)	Enron dataset,SMS Spam Collection Data Set	Naive Bayes,SVM, Decision Tree, Random Forest	TF-IDF	RF, Accuracy 97.6%
Olatunji (2019)	Spam Base	SVM	Not specified	Accuracy 94.06%
Ouyang et al. (2023)	Enron dataset	KNN, Naive Bayes	BoW	KNN, Accuracy 80%

PROPOSED METHODOLOGY

In this part, there are a few stages that are required to finalize the research. Total of 5 stages that the research consists of which are the data preparation, data preprocessing, data visualization, model implementation, and model evaluation.

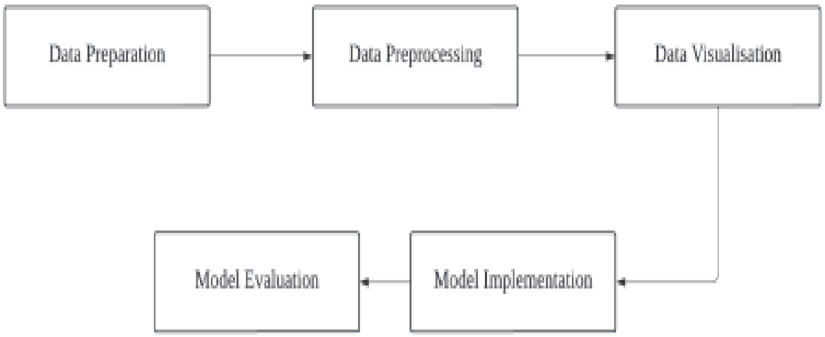


***Fig.1:**-Stages of Proposed Techniques*

Data Preparation

The proposed model worked on Python Language with its strong language and vast libraries packages available make it easier to develop. The dataset was procured from Kaggle, an open-source platform that stored a variety of datasets by researchers in AI projects [9] . This dataset is called the Enron Email Dataset which contains 5572 email entries. There are nearly 87% of ham email entries and 13% of spam email entries which are suitable for training machine learning algorithms. 2 columns are given, one showing the ham/spam label while another one showing the messages. Besides, we had also extracted the entries with charts shown in Figure 2.

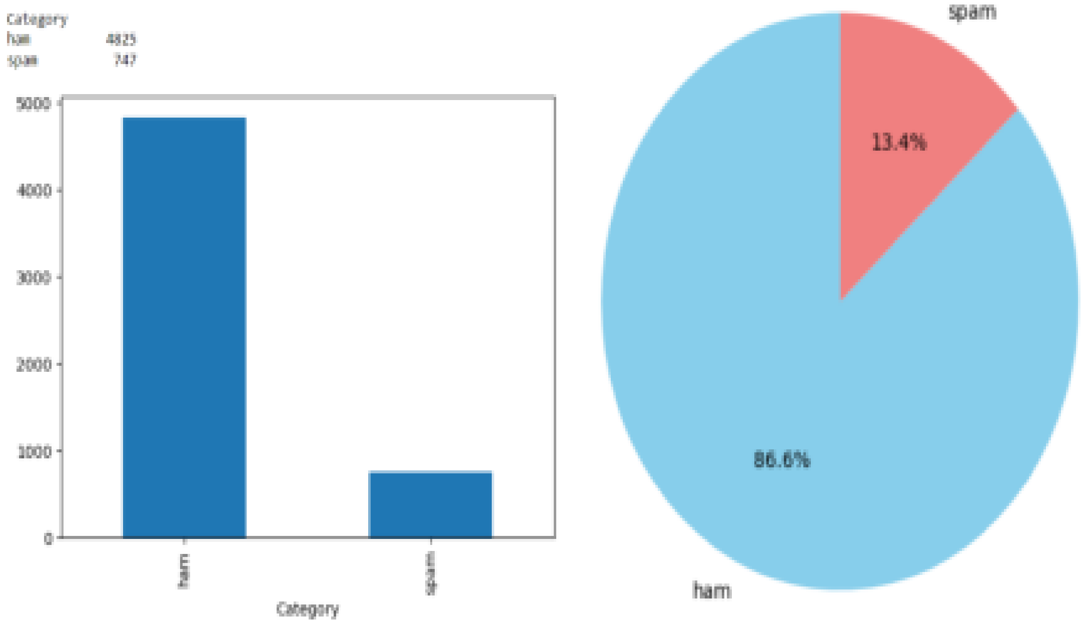


Fig.2:*-Distribution of ham and spam emails in bar and pie chart*

Data Preprocessing

Data preprocessing step is one of most important steps in NLP to extract the messages inside the dataset into proper and better words and serve as input for the algorithms to run with increasing accuracy and removing unimportant words. For example, we performed word tokenization, tokenizing every message from a string to word, and also stopwords removal, removing unnecessary words from the tokenized words as well as word lemmatization, lemmatizing the tokenized words to their natural form of structure. The reason we preferred lemmatization over stemming is because lemmatization returned more meaningful and important words than stemming which resulted in an accurate structure form of words to the algorithm.

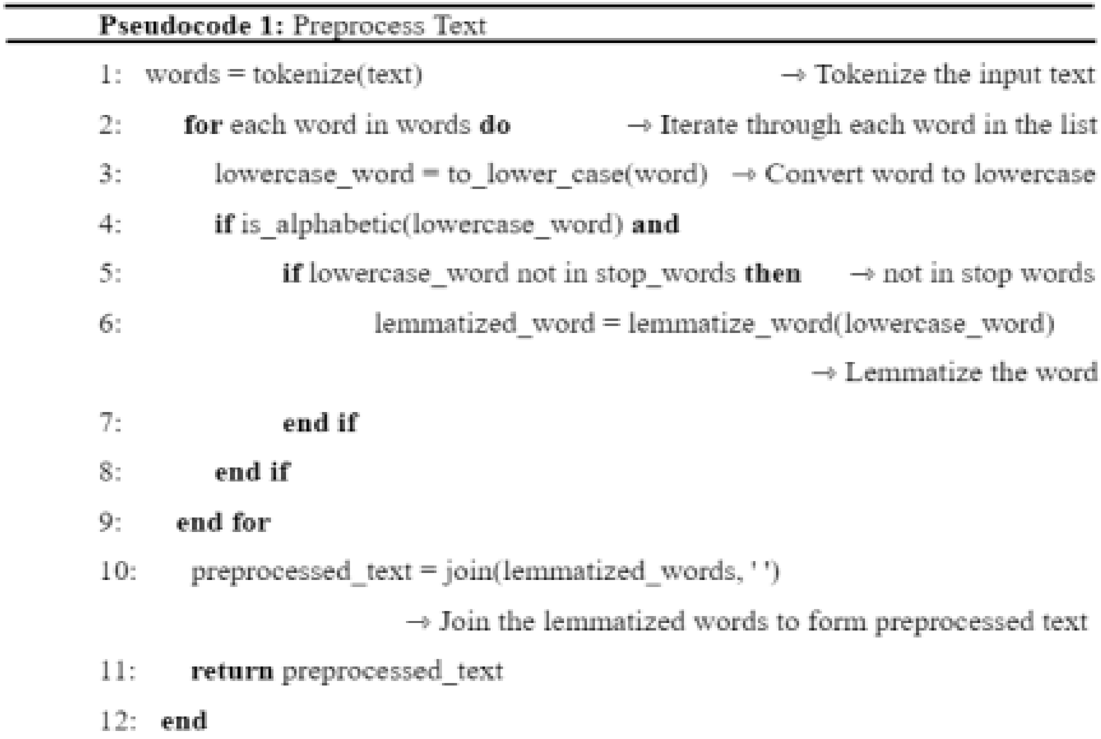
Data Visualisation

Data visualisation step is a step for us to understand well about the preprocessed text with the form of visualising image or picture. Of course, to visualise the data, we import the WordCloud library, to represent words into graphical representation. The visualisation was worked with the cleaned and preprocessed text, on both ham and spam messages.

From figure 2, we can see that specifically in spam emails, the frequency of “free” and “call” are holding the highest frequency over all the other words such as “text”, “mobile” and so on. Whereas on ham emails, “u”, “lt”, “gt” these words are having high frequency as well over the ham emails.

Model Implementation

As mentioned, this paper worked on 3 different models which are Naive Bayes, KNN and SVM. Before that, there are still a few steps to prepare.



```
Pseudocode 1: Preprocess Text
words = tokenize(text)                                  → Tokenize the input text
  for each word in words do                             → Iterate through each word in the list
    lowercase_word = to_lower_case(word)                → Convert word to lowercase
    if is_alphabetic(lowercase_word) and
      if lowercase_word not in stop_words then          → not in stop words
          lemmatized_word = lemmatize_word(lowercase_word)
                                                        → Lemmatize the word
      end if
    end if
  end for
  preprocessed_text = join(lemmatized_words, ' ')
                      → Join the lemmatized words to form preprocessed text
  return preprocessed_text
end
```

Fig.3:-Pseudocode for text preprocessing

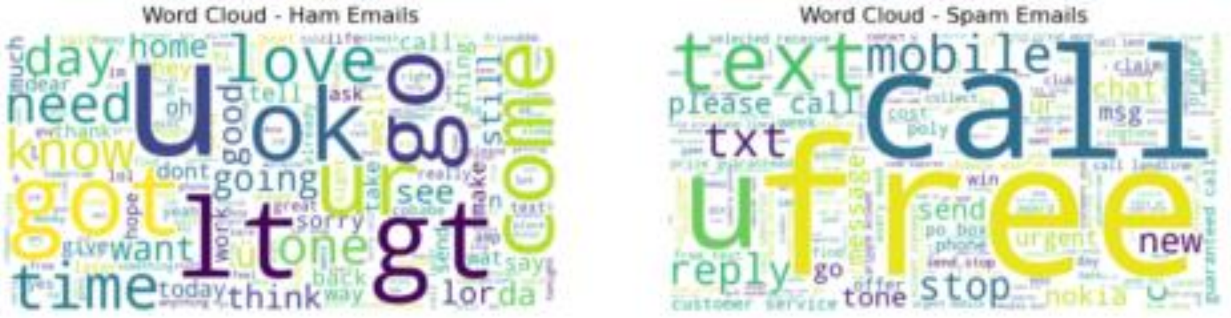


Fig.4:-WordCloud Results on Ham and Spam Emails

To serve the data as input, feature extraction is important to translate words to integers that can be understood by machine. For the feature extraction layer, we proposed the CountVectorizer and TfidfVectorizer that represent the Bag of Words (BoW) and Tf-idf techniques. To adopt these vectorizers, it is required to import the feature extractions from the scikit-learn library that provides statistical tools.

Data Splitting

Data splitting phase facilitates the division of the dataset into two distinct parts: the training set and the testing set. The training set plays a crucial role in assessing the effectiveness of machine learning techniques by utilizing data samples from the dataset. On the other hand, the testing set is employed to evaluate the performance of these techniques. The train and test functions were implemented to separate the two data categories. The dataset underwent a split, with 80% of the data allocated to the training set and the remaining 20% to the testing set. This uneven distribution of data samples ensures an unbiased performance percentage for spam classification. The random state created the randomness of splitting ham and spam messages evenly distributed to 80% training and 20% test set.

The Implementation of model classifications included Naive Bayes, K-Nearest Neighbour and Support Vector Machine, as aforementioned using both Bag of Words and Term Frequency Inverse Document Frequency approach. 6 different Machine Learning algorithms have been implemented in order to perform a comparative analysis of their performance and choose the model with the best performance for our proposed spam filtering through several metrics.

i. Naive Bayes Classifier + CountVectorizer
ii. Naive Bayes Classifier + TfidfVectorizer
iii. K-Nearest Neighbour Classifier + CountVectorizer
iv. K-Nearest Neighbour Classifier + TfidfVectorizer
v. Support Vector Machine Classifier + CountVectorizer
iv. Support Vector Machine Classifier + TfidfVectorizer

Model Evaluation

To determine the performance of implemented models, evaluation of model’s performance can be based on several metrics. Accuracy metric is the most used metric in many models however we are able to involve 3 others metrics which are Precision, Recall and F1-score.

$$A = \frac{\Sigma TP + \Sigma TN}{\Sigma All\ Testing\ Entries} \qquad (1)$$

$$P = \frac{\Sigma TP}{\Sigma TP + \Sigma FP} \qquad (2)$$

$$R = \frac{\Sigma TP}{\Sigma TP + \Sigma FN} \qquad (3)$$

$$F1 = \frac{R \times P}{R + P} \times 2 \qquad (4)$$

Confusion Matrix

The confusion matrix is a table that is used in machine learning and the assessment of a classification model’s performance involves utilising various statistical metrics. It provides a

Table 2:*-Notation for Metrics*

A	Accuracy
ΣTP	Number of True Positive Entries
ΣTN	Number of True Negative Entries
P	Precision
ΣFP	Number of False Positive Entries
R	Recall
ΣTP	Number of False Negative Entries
F1	F1-Score

comprehensive overview of a classification model’s performance by presenting the counts of four parameters which are true positive (TP), false positive (FP), true negative (TN), and false negative (TP). TP is the count of samples accurately predicted as positive. Meanwhile, the TN is the count of samples accurately predicted as negative. On the other hand, the FP is the count of samples incorrectly predicted as positive, while the FN is the count of samples incorrectly predicted as positive.

RESULTS AND DISCUSSIONS

We have imputed both train and test sets for the implemented model and analyse the results based on the evaluation metrics as detailed. We will also list out and compare the performance of the proposed models to determine the best model for spam filtering we have proposed that tally to our objectives.

***Table 3:**-Tabulated Summary Evaluation Results*

Algorithms		Naive Bayes		K-Nearest Neighbour (K-Value = 3)		Support Vector Machine	
		BoW	TFIDF	BoW	TFIDF	BoW	TFIDF
Precision	Ham	99%	97%	94%	93%	98%	99%
	Spam	91%	100%	99%	98%	100%	99%
Recall	Ham	99%	100%	100%	100%	100%	100%
	Spam	93%	79%	54%	44%	88%	90%
F1-score	Ham	99%	99%	97%	96%	99%	99%
	Spam	94%	88%	70%	61%	94%	94%
Accuracy		98.20%	97.48%	94.34%	93.18%	98.56%	98.65%

Table 3 showed the summary of evaluation results from 6 different proposed algorithms based on the evaluation metrics we have discussed before which are Precision, Recall, F1- score as well as accuracy. From the tabulated results, SVM has the highest accuracy value surpassing other algorithms with a value of 98.65%. While every scoring that SVM have showed higher than any other algorithms, Naive Bayes with TF-IDF feature extraction is actually a second-class compared to SVM as it has 100% of recall scoring on ham message classification which means there was not a single ham message is been misclassified as spam message. In general, SVM is the most well-performing algorithm based on the metric scoring it has is actually suitable for spam filtering.

Comparison between Proposed and Related Works

***Table 4:**-Tabulated Comparison between Proposed and Related Works*

Algorithms		Naive Bayes Chavez (2020)	RF Junnarkar et al (2021)	SVM Olatunji (2019)	KNN Ouyang et al (2023)	Proposed SVM with TFIDF
Precision	Ham	97%	98%	94.15%	76.84%	99%
	Spam	100%	97%	94.18%	80%	99%
Recall	Ham	100%	99%	95.83%	98.65%	100%
	Spam	80%	95%	95.60%	84.61%	90%
F1-score	Ham	99%	98%	94.98%	86.36%	99%
	Spam	89%	96%	94.88%	82.24%	94%
Accuracy		97.50%	97.60%	94.06%	80%	98.65%

From table 4 shown the overall accuracy of the proposed model is the highest accuracy across all other algorithms from related works. To further evaluate the performance, the other metrics including precision, recall and F1-score are carried out from the related works. The precision score of the proposed algorithm is the highest compared to other related algorithms with 99% on ham/spam classification.

For the recall of ham classification, the Naive Bayes algorithm from Chavez (2020) is having the same perfect 100% as ours did but the recall for spam classification is poorer than ours. Talking about recall for spam classification, both Random

Forest from Junnarkar et al. (2022) and SVM from Olatunji (2019) are performing better than ours which is around 95%. In F1-score for spam classification, the algorithm proposed by Junnarkar et al. (2021) is higher than ours since both input score from precision and recall was balanced.

CONCLUSION

This project aims to enhance artificial intelligence in the field of machine learning by primarily focusing on developing efficient spam filtering techniques to combat the issue of spam emails.

The study investigates several approaches of extracting features by utilizing natural language processing techniques and a supervised learning approach with the Enron dataset. The Support Vector Machine (SVM) algorithm, when combined with the Term Frequency Inverse Document Frequency (TF-IDF) technique, proves to be the most efficient approach, outperforming other algorithms in terms of performance measures. This project enhances the development of spam filtering by focusing on precision, recall, and F1-score. It provides a comprehensive machine learning-based solution to address the important cybersecurity issue of spam.

REFERENCES

1. R. K. Karn, V. E. Jesi, S. M. Aslam, Spam Email Detection Using Machine Learning Integrated In Cloud, 2023 International Conference on Networking and Communications (ICNWC) (2023) 1–8.
2. S. Rao, A. K. Verma, T. Bhatia, A review on social spam detection: challenges, open issues, and future directions, Expert Systems with Applications 186 (2021) 115742–115742.
3. S. K. Reddy, T. Padmaja, Maruthi, Non Machine and Machine Learning Spam Filtering Techniques, International Journal of Recent Technology and Engineering (IJRTE) (7) (2019) 2277–3878.
4. E. G. Dada, J. S. Bassi, H. Chiroma, A. O. Adetunmbi, O. E. Ajibuwa, Machine learning for email spam filtering: review, approaches and open research problems, Heliyon (6) (2019) 5–5.
5. A. Chavez, TF-IDF classification based Multinomial Naïve Bayes model for spam filtering (Doctoral dissertation, Dublin, National College of Ireland, 2020.
6. A. Junnarkar, S. Adhikari, J. Fagania, P. Chimurkar, D. Karia, E-mail spam classification via machine learning and natural language processing, 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV) (2021) 693–699.
7. S. O. Olatunji, Improved email spam detection model based on support vector machines, *Neural Computing and Applications* 31 (2019) 691–699.
8. Q. Ouyang, J. Tian, J. Wei, E-mail Spam Classification using KNN and Naive Bayes, *Highlights in Science, Engineering and Technology* 38 (2023) 57–63.
9. A. Mohinur, The Enron Email Dataset (2022).URL https://www.kaggle.com/datasets/mohinurabdurahimova/maildataset/data *January 19, 2024*