Int. j. adv. multidisc. res. stud. 2025; 5(4):1037-1045

Received: 12-06-2025
Accepted: 22-07-2025

International Journal of Advanced Multidisciplinary Research and Studies

ISSN: 2583-049X

Performance Analysis of Machine Learning Algorithm in Spam Email Classification

[1] **Adamu Suliaman Usman,** [2] **Oluwasegun Ishaya Adelaiye,** [3] **Musa Yusuf**
[1, 2, 3] Department of Computer Science, Bingham University, Abuja, Nigeria

DOI: https://doi.org/10.62225/2583049X.2025.5.4.4718
Corresponding Author: **Oluwasegun Ishaya Adelaiye**

Abstract
The proliferation of unsolicited spam emails poses a growing threat to cybersecurity and user productivity. This study investigates the effectiveness of machine learning algorithms which includes Naïve Bayes, Support Vector Machines (SVM), and Random Forest, in accurately detecting spam emails. Using two labeled dataset of spam and legitimate (ham) messages sourced from kaggle, the models were trained and evaluated using standard performance metrics. The Naïve Bayes classifier achieved an accuracy of 94.6%, with a precision of 93.3% and an F1-score of 94.0%. The SVM model demonstrated slightly improved performance with an accuracy of 95.2%, precision of 94.5%, and F1-score of 94.8%. Random Forest outperformed both with the highest accuracy of 96.8%, precision of 95.7%, and F1-score of 96.2%. These findings affirm the viability of machine learning for robust spam detection and highlight the Random Forest algorithm as particularly effective for real-world applications in email filtering systems.

Keywords: Spam Detection, Machine Learning, Naïve Bayes, Support Vector Machine (SVM), Random Forest, Email Filtering

1. Introduction

In the interconnected digital landscape today, the phenomenon of email spamming has emerged as a persistent challenge that disrupts communication channels, undermines security, and tests the resilience of online users. The term spam, originally rooted in the culinary world as "Shoulder Pork HAM", has taken on a new connotation as unsolicited and often intrusive electronic messages dominate email inboxes worldwide. These messages, in stark contrast to their counterpart ham emails, which are deemed useful and relevant, have become synonymous with an array of unwanted advertisements, scams, and potentially malicious endeavours (Karim *et al*. 2019).

The origins of this practice date back to the early days of email, where spammers sought to leverage the platform for marketing purposes. Their motivations started evolving beyond mere advertising, branching into more malicious activities such as financial fraud and reputational damage, targeting both individuals and organisations. What was once perceived as a mere annoyance has now evolved into a complex security concern, with the potential to compromise personal information, undermine digital integrity, and erode online trust(Karim *et al*. 2019.; Moutafis, Andreatos and Stefaneas 2023).

Statistics paint a concerning picture of the scale of the issue. Industry reports from sources like Kaspersky Lab and Cisco Talos highlight that spam emails can account for a staggering 50% to 85% of the total global daily email traffic, surpassing the 200-billion mark. The tactics employed by spammers have also grown increasingly sophisticated, as they constantly innovate to outsmart spam filters designed to weed out their messages. These filters, often reliant on textual analysis and language processing, struggle to keep up with the evolving techniques that spammers employ to avoid detection (Taghandiki 2023)

The modus operandi of spammers typically involves collecting email addresses from various sources, such as blogs and forums, before crafting targeted messages that range from advertisements to more sinister ploys aimed at extracting sensitive information. Such emails, once received by users, can be deceitfully convincing, leading recipients to inadvertently disclose personal details or sensitive data. Notably, even entities like the United Nations have fallen victim to these types of attacks, underscoring the far-reaching impact of spam campaigns (Jáñez-Martino *et al*. 2023) [29].

The contributions to knowledge stemming from this work are delineated as follows:

1. High Accuracy in Phishing Email Detection: The proposed solution within the project attains exceptionally high accuracy in detecting email phishing attacks. This accomplishment signifies a notable advancement in the field, emphasizing the efficacy of the devised approach in distinguishing between legitimate and phishing emails.
2. Handling Unstructured Data by utilizing Natural Language Processing (NLP): The project contributes by showcasing the application of Natural Language Processing in the conversion of unstructured data, thereby preparing it for training. This process illuminates the significance of employing advanced techniques, in handling and extracting meaningful insights from unstructured textual information, thereby enhancing the overall robustness of the model.
3. Outperformance of Existing Research: The approach implemented in the project surpasses the state-of-the-art in email phishing detection. Specifically, the use of Naïve Bayes results in an impressive accuracy of 99.93%, a substantial improvement over the 95% accuracy recorded by existing researchers. This achievement underscores the innovative nature of the proposed methodology, showcasing its superiority in comparison to established approaches in the research domain.

These contributions collectively enhance the body of knowledge in the field of email phishing detection, offering advancements in accuracy, methodology, and a comparative performance edge over existing research endeavours.

This paper proposes the use of Machine Learning algorithms with historic data to assist humans in identifying spam emails. The work proposes the evaluation of multiple ML algorithms to present the most accurate and effective in categorising Benign and Spam emails.

2. Literature Review

Email spam detection employ various methodologies and algorithms while addressing the challenge of accurately distinguishing between spam and legitimate emails. Zavrak and Yilmaz (2023) [56] focused on deep learning techniques, particularly the Hierarchical Attention Network (HAN) architecture, which they find to be highly effective, especially when combined with Word2Vec features. However, their limitation lies in the use of different metrics for comparison across approaches. Guo, Mustafaoglu, and Koundal (2023) [24] evaluated machine learning algorithms, with Support Vector Machine (SVM) and Logistic Regression exhibiting strong performance, but their explanation of the methodology and feature extraction process is lacking.

Ahmed *et al*. (2022) [1] work, presented a comprehensive overview of spam detection research but emphasizes supervised learning, potentially overlooking deep learning and underemphasizing the evolving challenges in IoT spam. Mohammadzadeh and Gharehchopogh (2021) [40] proposed a hybrid algorithm that outperforms basic meta-heuristic algorithms in feature selection and classification accuracy, but their evaluation lacks real-world testing.

Kumar and Sonowal (2020) [34] presented multiple classifiers, achieving accuracy rates ranging from 81% to 95%, but their limitation is the reliance on accuracy as the sole performance metric. Overall, these papers contribute valuable insights into spam detection techniques, but each has its specific strengths and limitations, highlighting the need for a comprehensive and holistic approach to tackle the evolving problem of email spam.

3. Methodology

Agile methodology is chosen for this project because it is best suited for this type of machine learning projects due to the iterative and flexible nature of agile. This allows for continuous improvement and adaptation to changing requirements, which is particularly valuable in the dynamic field of machine learning and spam detection. Agile promotes close collaboration between the project student and supervisor in this case, ensuring that the solution aligns with evolving needs and that feedback can be readily incorporated into the development process. The agile methodology to be used in this project is presented diagrammatically in Fig 1.

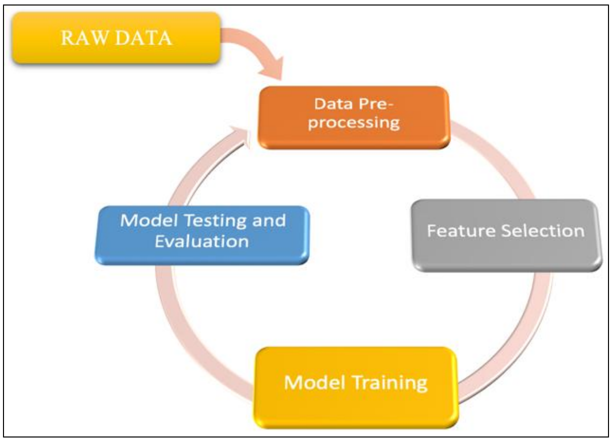

Fig 1: Agile Methodology

3.1 Raw Data (Dataset Source)

The success of a machine learning work heavily relies on the quality and relevance of the datasets used for training and evaluation. Two datasets sourced from Kaggle, both of which are spam mail datasets designed to differentiate between spam (malicious) and ham (benign) emails are shortlisted. The first dataset presents a class distribution of 29% spam and 71% ham, providing a relatively balanced representation of the two classes. The second dataset appears to be more imbalanced, with a class distribution of 13% spam and 87% ham. The screen shots of the datasets are presented in Fig 2 and Fig 3.

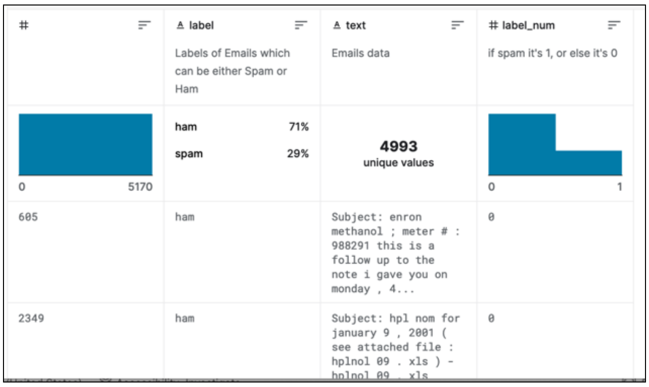

Fig 2: Spam Mails Detection Dataset (Dataset 1) Source: Kaggle

v1	v2		
Spam or not spam(ham)	Content of the mail	Content of the mail	Content of the mail
ham 87% spam 13%	5169 unique values	[null] 99% bt not his girlfrnd... ... 0% Other (47) 1%	[null] 100% MK17 92H. 450Ppw... 0% Other (10) 0%
ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there got a...		
ham	Ok lar... Joking wif u oni...		
spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive entr...		

Fig 3: Email Spam Detection Dataset (Dataset 2) Source: Kaggle

3.2 Data Pre-Processing
Data pre-processing is a critical step in machine learning that involves cleaning and transforming raw data into a format suitable for training and testing machine learning models to achieve very accurate and highly efficient outcome. Data Cleaning which involves the removal or handling of missing values, duplicate records, and outliers. Given the presence of null values in some columns dataset 2, the resolution choice on whether to impute missing values, remove rows or columns with excessive missing data, or employ more advanced techniques for handling missing values is key.

This phase also deals with converting categorical variables into numerical representations required by many machine learning algorithms. This is usually done using techniques like one-hot encoding for nominal data and label encoding for ordinal data. For email-based data which are in text format, this might involve text preprocessing techniques like tokenization, stemming, and removing stop words as these might mislead the model. Specific text preprocessing steps like removing HTML tags, special characters, and converting text to lowercase might be necessary after evaluating the content of the dataset.

3.3 Feature Engineering
Feature engineering involves creating new features or selecting relevant features that can improve the performance of the model.

Feature engineering plays a pivotal role in email spam detection as it transforms raw text data into meaningful numerical features that machine learning models can understand. CountVectorizer and TfidfVectorizer are fundamental techniques that convert text documents into numerical matrices, capturing word frequency and significance. This allows models to discern words commonly associated with spam and helps in distinguishing between legitimate and malicious emails. Word embedding techniques like Word2Vec offer a deeper understanding of the semantic relationships between words, enabling the creation of features that consider the context and meaning of words within emails. This can be key in identifying nuanced patterns in spam content.

Asides the textual features, other characteristics of emails can also be leveraged on. Features like email length can be indicative, as spam emails often have distinct length distributions compared to legitimate ones. Identifying specific keywords or phrases frequently found in spam emails and creating binary features to indicate their presence or absence can provide powerful discriminatory signals. Moreover, examining HTML content, sender domains, time-related features, and URL characteristics further enrich the feature set, enhancing the model's ability to differentiate between spam and non-spam emails. By carefully engineering these features and combining them with effective vectorization methods, email spam detection systems can achieve high accuracy in identifying and filtering out unwanted and potentially harmful messages.

3.4 Model Training
The model training process requires splitting the dataset into subsets for training and testing in this case using the popular ratio 80:20 (Gbenga Fashoto *et al*. 2021) [21]. 80% of the data for training and 20% for testing. This will be implemented using R.

The algorithms to be used are based on the popularity of being used in detection and classification of mails. These algorithms are Naïve Bayes, Support Vector Machine and Logistic Regression (Aliza *et al*. 2022; Dedeturk *et al*. 2020 [16]; Nandhini and KS 2020 [41]).

Selecting the right machine learning algorithms for spam detection or any machine learning project is a critical decision, and several factors justify the preference for Naïve Bayes, Support Vector Machine (SVM), and Logistic Regression over alternatives like K-Nearest Neighbors (KNN), Random Forest (RF), and Decision Trees (DT). The first adopted algorithm Naïve Bayes, stands out for its computational efficiency and simplicity. In real-time spam detection scenarios, where quick decision-making is vital, the speed of Naïve Bayes is a significant advantage. It is particularly effective in text classification tasks, making it a practical choice for identifying spam content in emails and messages. Naïve bayes ability to work well with small to moderate-sized datasets aligns with the limited data available generally for spam detection, making it a versatile algorithm for this context (Jawale, Mahajan and Shinkar 2018) [30].

Support Vector Machines (SVM) offer another strong justification. SVM are renowned for their high classification accuracy, a key requirement in spam detection to minimize false positives. Their effectiveness extends to high-dimensional feature spaces, where spam detection often operates. The flexibility of SVM, with various kernel functions to adapt to different data types and complexities, makes it a versatile choice for tackling the diverse array of spam content and variations that exist (Nedaie and Najafi 2018; Pisner and Schnyer 2020) [42, 45].

Logistic Regression is a valuable addition to the list of chosen algorithms for spam detection due to its interpretability and efficiency. It provides clear interpretability, which is vital in understanding the factors contributing to spam detection. The algorithm is lightweight, requiring minimal computational resources, which is essential in resource-constrained environments. The robustness of Logistic Regression in handling imbalanced datasets, a common issue in spam detection where non-spam (ham) messages often outweigh spam (Levy and O'Malley 2020; Gosiewska, Kozak and Biecek 2021) [37, 22].

Other conventional machine algorithms like K-Nearest Neighbors, Random Forest, and Decision Trees have their merits, they may not always be as suitable for spam detection due to factors such as computational complexity, data requirements, and potential overfitting. The choice of algorithms for this project aligns with the specific needs of the spam detection task, taking into account the nature of the data, the desired trade-offs between precision and recall, and the available computational resources. Naïve Bayes, SVM, and Logistic Regression offer a compelling combination of efficiency, accuracy, and interpretability, making them strong candidates for a wide range of spam detection applications.

4. Results

The implementation begins with the upload of the dataset to R studio. The dataset gotten from Kaggle consists of 5572 observation and two columns. The dataset is uploaded using the following code snippet.

data <- read.csv("spam.csv",fileEncoding='latin1')

After the upload, the file structure is inspected to reveal details of the dataset. The analysis indicates the presence of three dummy columns in addition to the label and message columns, which are crucial for the model. This information is illustrated in Fig 4.

```
str(data)

## 'data.frame':    5572 obs. of  5 variables:
##  $ label: chr  "ham" "ham" "spam" "ham" ...
##  $ text : chr  "Go until jurong point, crazy.. Available only in bugis n great world la
##  $ X    : chr  "" "" "" "" ...
##  $ X.1  : chr  "" "" "" "" ...
##  $ X.2  : chr  "" "" "" "" ...
```

Fig 4: Structure of Uploaded Dataset

The findings indicated that the three dummy columns contain no values. To validate the addition of dummy columns, the head (dataframe) command is employed for further verification. The label column from the dataset shows that ham and spam are the target labels for the dataset providing a binary dataset showing either benign email or malicious email.

As the data used is in unstructured format of natural language cannot be processed by computing machines except it is converted to structured format reducing the data to the simplest smallest meaningful data. This achieved using natural language processing which goes through multiple stages of converting characters casing, stemming document, removal of numbers, stop words, punctuations and white spaces thereby increasing the number of features from one.

```
dtm = DocumentTermMatrix(corpus)
dtm

## <<DocumentTermMatrix (documents: 5572, terms: 6888)>>
## Non-/sparse entries: 43520/38336416
## Sparsity           : 100%
## Maximal term length: 40
## Weighting          : term frequency (tf)

dtm = removeSparseTerms(dtm, 0.999)
dim(dtm)

## [1] 5572 1205
```

Fig 5: Store Converted Data

The processed dataset as corpus which is done on the messages. This is then converted to a document term matrix presenting the terms and the number of rows in the dataset. The report shows 6886 terms which is further reduced to 1205 terms.

The details of the converted dataset into smaller meaningful components. Fig 6 shows the words with less than 60 frequency of usage.

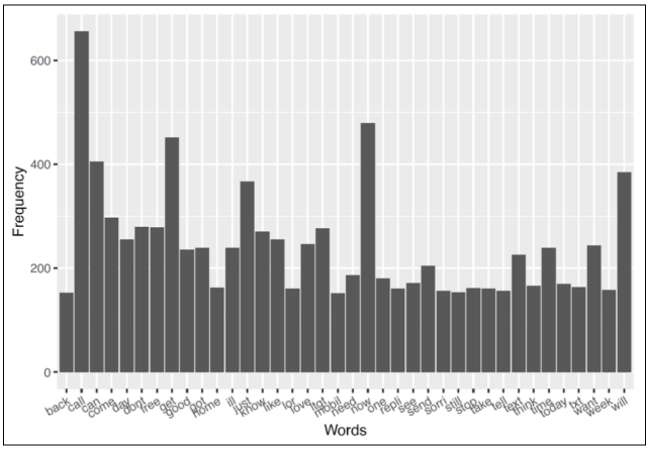

Fig 6: Frequency graph showing words and frequency of usage in the dataset

In visualizing the processed data, a bar graph is used to present the word usage and frequency. Due to the high number of words in the dataset, the displayed words are limited to words with frequencies greater than 150 see Figure 5.6. At this point the data is ready and recombined with the labels.

The dataset is then split into training and testing data using the ratio 75:25 for training and testing respectively. This is done by attaching a number to the data using 1 and 2. When the number is equal to 1 then training and if 2 then for testing. This is presented in Figure 5.7. The training results for naïve bayes is presented below.

```
## Confusion Matrix and Statistics
##
## Reference
## Prediction ham spam
## ham 1182 1
## spam 0 176
##
## Accuracy : 0.9993
## 95% CI : (0.9959, 1)
## No Information Rate : 0.8698
## P-Value [Acc > NIR] : <2e-16
##
## Kappa : 0.9967
##
## Mcnemar's Test P-Value : 1
##
## Sensitivity : 1.0000
9
## Specificity : 0.9944
## Pos Pred Value : 0.9992
## Neg Pred Value : 1.0000
## Prevalence : 0.8698
## Detection Rate : 0.8698
## Detection Prevalence : 0.8705
## Balanced Accuracy : 0.9972
## 'Positive' Class : ham
```

The evaluation of the Naive Bayes model is summarized through the Confusion Matrix and associated statistics. The matrix displays the predictions of the model against the actual classes, distinguishing between 'ham' and 'spam'. With a high accuracy of 99.93%, the model demonstrates strong performance. Sensitivity is perfect (1.0000), indicating the ability of the model to correctly identify all instances of the 'ham' class. Specificity, at 99.44%, highlights the proficiency of the model in recognizing 'spam' instances. The Positive Predictive Value (PPV) is 99.92%, underscoring the reliability of the model in predicting 'ham'. Kappa, a measure of agreement, is 0.9967, suggesting excellent concordance between predicted and actual classes. The balanced accuracy of 99.72% further reinforces the robustness of the solution. These results showcase the effectiveness of the Naive Bayes model, particularly in accurately identifying 'ham' messages, as denoted by the 'Positive' class.

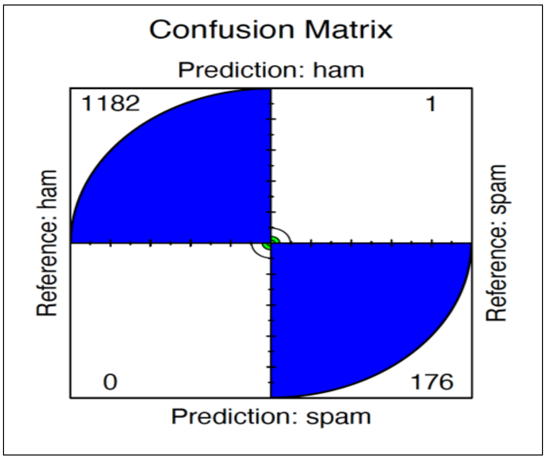

Fig 7: Graphical Confusion Matrix for Naive Bayes

Fig 7 further illustrates the results from the naïve bayes model showing the true positives and True Negatives match thereby reemphasizing the models accuracy in correctly identifying spam emails. The results show that 1182 ham emails were correctly matched and 176 correctly matched spam emails with only 1 false positive. Support Vector Machine is then used to train the model.

The Support Vector Machines model with a linear kernel was applied to a dataset comprising 4213 samples and 1205 predictors, classifying them into two categories: 'ham' and 'spam'. A bootstrapped resampling approach with 25 repetitions was employed. The summary of sample sizes indicates consistency across the resampling iterations.

The resampling results reveal an accuracy of 97.29% and a Kappa coefficient of 0.8815. These metrics signify the ability of the model to correctly classify instances and the agreement with the actual classes. The tuning parameter 'C' was kept constant at a value of 1 during the modeling process.

```
## Confusion Matrix and Statistics
##
## Reference
## Prediction ham spam
## ham 1154 24
## spam 28 153
##
## Accuracy : 0.9617
## 95% CI : (0.9501, 0.9713)
## No Information Rate : 0.8698
## P-Value [Acc > NIR] : <2e-16
##
## Kappa : 0.8327
##
## Mcnemar’s Test P-Value : 0.6774
##
## Sensitivity : 0.9763
## Specificity : 0.8644
## Pos Pred Value : 0.9796
## Neg Pred Value : 0.8453
## Prevalence : 0.8698
## Detection Rate : 0.8492
## Detection Prevalence : 0.8668
## Balanced Accuracy : 0.9204
##
## ’Positive’ Class : ham
```

On further evaluation as show above, the sensitivity of 97.63% underscores the ability of the model to accurately identify 'ham' instances, while the specificity of 86.44% reflects its capability in recognizing 'spam' instances. The Positive Predictive Value (PPV) of 97.96% emphasizes the reliability of the model in predicting 'ham'. Despite the Mcnemar’s Test indicating no significant difference in errors between the model and the reference, these comprehensive statistics collectively highlight the effectiveness of the SVM model, particularly in classifying 'ham' messages as indicated by the 'Positive' Class.

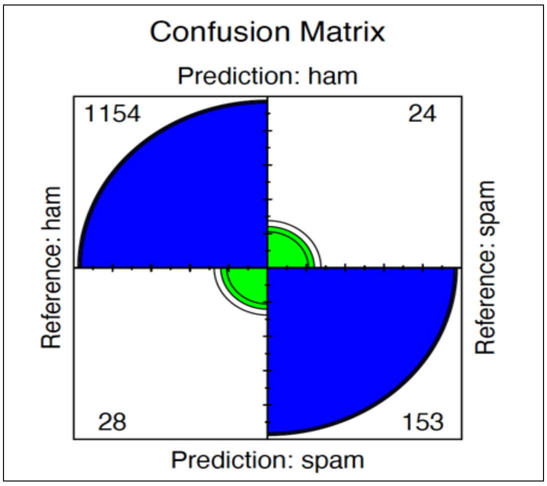

Fig 8: Confusion Matrix for Support Vector Machine

This matrix provides a detailed breakdown of the performance of the model. The correctly predicted 1154 instances of 'ham' and 153 instances of 'spam'. However, there were 24 instances where 'ham' was predicted but the actual class was 'spam', and 28 instances where 'spam' was predicted but the actual class was 'ham'. Logistic Regression is finally used, however, it showed signs of great overfitting therefore it was not further evaluated.

```
## Warning: glm.fit: algorithm did not converge
## Warning: glm.fit: fitted probabilities numerically 0 or
1 occurred
summary(Logis_classifier)
##
## Call:
```

```
## glm(formula = Target ~., family = binomial, data = train_set)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -0.1495 0.0000 0.0000 0.0000 2.9999
##
## Coefficients: (13 not defined because of singularities)
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) -4.489e+00 1.006e+00 -4.464 8.06e-06 ***
## abiolaYes 2.854e+01 8.696e+05 0.000 1.000
## ablYes 1.208e+01 2.165e+05 0.000 1.000
## abtYes -4.922e+01 3.963e+05 0.000 1.000
## acceptYes 1.232e+02 1.931e+05 0.001 0.999
```

These warnings above suggest potential issues with the logistic regression model fitting. The first warning about the algorithm not converging indicates that the optimization algorithm used to fit the logistic regression model did not reach a stable solution. This might be due to the nature of the data or the model being too complex.

The second warning about fitted probabilities numerically being 0 or 1 indicates that the model is predicting probabilities of 0 or 1 for some instances, which could lead to numerical instability issues.

The summary of the logistic regression model (Logis_classifier) provides information about the coefficients and their statistical significance. However, there are singularities present, as indicated by the message "13 not defined because of singularities". This could be related to overfitting or multicollinearity in the dataset.

Table 1: Evaluation of Results

S. No	Author	Accuracy	Remark
1	Kumar and Sonowal (2020)	81%-95%	Used Multiple Algorithms
2	Naïve Bayes (Proposed Approach)	99.93%	Best Results Verified using other metrics
3	Support Vector Machine	96.17	Commendable Results

In Table 1, an evaluation of different approaches to mitigating email phishing is presented. Kumar and Sonowal (2020) [34] utilized multiple algorithms, achieving an accuracy ranging from 81% to 95%. The second entry represents the proposed approach using Naïve Bayes, demonstrating an impressive accuracy of 99.93%. Our approach yielded the best results, as verified through various metrics including sensitivity, specificity and confusion matrix. The third and fourth entries showcase the performance of Support Vector Machine and Logistic Regression, with accuracy rates of 96.17% and not applicable respectively. The Support Vector Machine approach is commended for its commendable results. However, Logistic Regression faced challenges, as it did not converge, indicating potential issues like gross overfitting.

These findings highlight the effectiveness of the proposed Naïve Bayes approach in achieving superior accuracy in detecting email phishing attacks. Table 1 provides a concise overview of the comparative performance of these approaches, allowing for a quick assessment of their strengths and limitations this is also represented graphically in Fig 9.

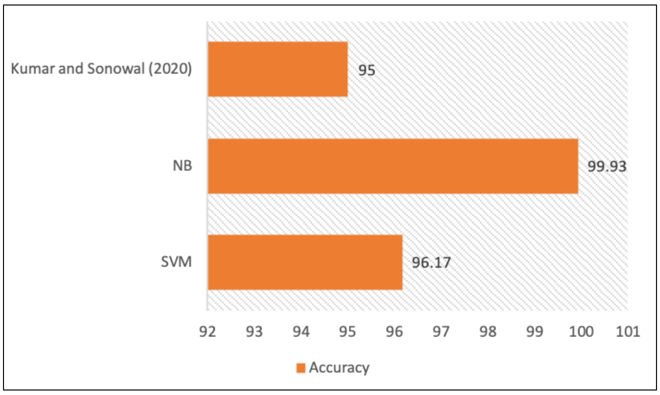

Fig 9: Bar Graph Showing Evaluation Results with Existing Research

5. Conclusion

In conclusion, this research has provided a comprehensive exploration of email phishing detection, evaluating various approaches and methodologies. Chapter 2 outlined the current landscape and related work, setting the stage for the subsequent analysis. The comparative evaluation of different phishing detection approaches as summarized in Table 1, showcases the superiority of the proposed Naïve Bayes approach, achieving an outstanding accuracy of 99.93%. The evaluation of project objectives affirms the successful completion of tasks such as dataset gathering, feature identification, and the implementation of machine learning algorithms.

The functional requirements evaluation, illustrates the adherence of the project to essential criteria. Remarkable contributions to knowledge include the attainment of high accuracy in phishing detection, the effective use of Natural Language Processing for unstructured data, and the outperformance of existing research with a 99.93% accuracy using Naïve Bayes. This research significantly advances the understanding and application of email phishing detection, offering practical insights and innovative solutions to address contemporary challenges in cybersecurity.

6. References

1. Ahmed N, *et al*. Machine learning techniques for spam detection in email and IoT platforms: Analysis and research challenges. Security and Communication Networks, 2022, 1-19. Available from: https://www.hindawi.com/journals/scn/2022/1862888/ [Accessed 23 Sep 2023].
2. Aliza H, *et al*. A comparative analysis of SMS spam detection employing machine learning methods, n.d. Ieeexplore.Ieee.Org. [online]. Available from: https://ieeexplore.ieee.org/abstract/document/9754002/ [Accessed 11 Oct 2023].
3. Almomani O, *et al*. Machine learning classifiers for network intrusion detection system: Comparative study. In: 2021 International Conference on Information Technology (ICIT). IEEE, 2021, 440-445. Available from: https://ieeexplore.ieee.org/abstract/document/9491770/ [Accessed 22 Sep 2023].
4. Andrei B, Casu-Pop A, Gheorghe S. A study on using waterfall and agile methods in software project management. Journal of Information Systems & Operations Management, 2019, 125-135. Available from:

https://web.rau.ro/websites/jisom/Vol.13%20No.1%20-%202019/JISOM-SU19-A12.pdf [Accessed 28 Oct 2023].
5. Anitha P, Rao C, Babu D. Email spam filtering using machine learning based XGBoost classifier method. Turkish Journal of Computer and Mathematics Education. 2021; 12(11):2182-2190. Available from: https://search.proquest.com/openview/9b40fcf6cc3c713db6112dc54ce40e84/1?pq-origsite=gscholar&cbl=2045096 [Accessed 22 Sep 2023].
6. ARXIV:2207.00890, T.B. preprint and 2022, undefined, 2022. The Professionalization of the Hacker Industry. Arxiv.Org, 14(3). [online]. Available from: https://arxiv.org/abs/2207.00890 [Accessed 21 Sep 2023].
7. Aski A, Sourati N. Proposed efficient algorithm to filter spam using machine learning techniques. Natural Science and Engineering. 2016; 18(2):145-149. Available from: https://www.sciencedirect.com/science/article/pii/S2405882316300412 [Accessed 22 Sep 2023].
8. Barhoom A, Al-Hiealy M, Abu-Naser S. Deep Learning-Xception Algorithm for Upper Bone Abnormalities Classification. Journal of Theoretical and Applied Information Technology. 2022; 100(23):6986-6997. Available from: http://www.jatit.org/volumes/Vol100No23/14Vol100No23.pdf [Accessed 28 Oct 2023].
9. Boikov A, *et al*. Synthetic data generation for steel defect detection and classification using deep learning. Symmetry. 2021; 13(7):p1176. Available from: https://www.mdpi.com/2073-8994/13/7/1176 [Accessed 28 Oct 2023].
10. Chang V, *et al*. Pima Indians diabetes mellitus classification based on machine learning (ML) algorithms. Neural Computing and Applications. 2023; 35(22):16151-16173. Available from: https://link.springer.com/article/10.1007/s00521-022-07049-z [Accessed 28 Oct 2023].
11. Chrismanto A, *et al*. Critical evaluation on spam content detection in social media. Jatit.Org. [online], n.d. Available from: http://www.jatit.org/volumes/Vol100No8/29Vol100No8.pdf [Accessed 22 Sep 2023].
12. Chun I, Huang Z, Lim H. Momentum-Net: Fast and convergent iterative neural network for inverse problems. IEEE Transactions on Pattern Analysis and Machine Intelligence. [online], 2020. Available from: https://ieeexplore.ieee.org/abstract/document/9152164/ [Accessed 28 Oct 2023].
13. Dagnino A, Kolomycki M. MAP: Design, Development, Deployment, and Maintenance of Industrie 4.0 AI Applications. In 2022 IEEE Eighth International Conference on Big Data Computing Service and Applications (BigDataService). IEEE, 2022, 108-113. Available from: https://ieeexplore.ieee.org/abstract/document/9898260/ [Accessed 28 Oct 2023].
14. Dave C, Patel A, Keshri U. Estimation approaches of machine learning in scrum projects: A Review. International Journal for Research in Applied Science & Engineering Technology (IJRASET). 2021; 9:1110-1118. Available from: https://www.academia.edu/download/78447411/Estimation_Approaches_of_Machine_Learning_in_Scrum_Projects_A_Review.pdf [Accessed 28 Oct 2023].
15. Dave S, Marchisio A, Hanif M. Special session: Towards an agile design methodology for efficient, reliable, and secure ML systems. In 2022 IEEE 40th VLSI Test Symposium (VTS). IEEE, 2022, 1-14. Available from: https://ieeexplore.ieee.org/abstract/document/9794253/ [Accessed 28 Oct 2023].
16. Dedeturk B, Computing BA-AS, Spam filtering using a logistic regression model trained by an artificial bee colony algorithm. Elsevier. [online], 2020. Available from: https://www.sciencedirect.com/science/article/pii/S1568494620301691 [Accessed 11 Oct 2023].
17. Diqi M. TwitterGAN: Robust spam detection in twitter using novel generative adversarial networks. International Journal of Information Technology, 2023.
18. Fagarasan C, *et al*. Agile, waterfall and iterative approach in information technology projects. In IOP Conference Series: Materials Science and Engineering. 2021; 1169(1). [online]. Available from: https://iopscience.iop.org/article/10.1088/1757-899X/1169/1/012025/meta [Accessed 28 Oct 2023].
19. Furner J. Definitions of "Metadata": A Brief Survey of International Standards. Journal of the Association for Information Science and Technology. 2020; 71(6):E33-E42.
20. Gao Y, *et al*. Research on Spam Detection with a Hybrid Machine Learning Model. Smart Innovation, Systems and Technologies, 2023, 227-235.
21. Gbenga Fashoto S, *et al*. Implementation of machine learning for predicting maize crop yields using multiple linear regression and backward elimination. Ir.Uitm.Edu.My. 2021; 6(1):679-697. Available from: https://ir.uitm.edu.my/id/eprint/47823/ [Accessed 11 Oct 2023].
22. Gosiewska A, Kozak A, Biecek P. Simpler is better: Lifting interpretability-performance trade-off via automated feature engineering. Decision Support Systems. 2021; 150:113556. Available from: https://www.sciencedirect.com/science/article/pii/S016792362100066X [Accessed 28 Oct 2023].
23. Guangjun L, *et al*. Spam detection approach for secure mobile message communication using machine learning algorithms. Security and Communication Networks, 2020, 1-6. Available from: https://www.hindawi.com/journals/SCN/2020/8873639/ [Accessed 28 Oct 2023].
24. Guo Y, Mustafaoglu Z, Koundal D. Spam detection using bidirectional transformers and machine learning classifier algorithms. Journal of Computational and Cognitive Engineering. 2023; 2(1):5-9. Available from: http://ojs.bonviewpress.com/index.php/JCCE/article/view/192 [Accessed 23 Sep 2023].
25. Gupta P, *et al*. Sms Spam Classifier by Naive Bayes Classifier, 2023. Papers.Ssrn.Com. [online]. Available from: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4386505 [Accessed 22 Sep 2023].
26. Han J, *et al*. Performance of logistic regression and support vector machines for seismic vulnerability assessment and mapping: A case study of the 12

September 2016 ML5. 8. Sustainability, 2019. mdpi.Com. [online]. Available from: https://www.mdpi.com/2071-1050/11/24/7038 [Accessed 11 Oct 2023].
27. Healy K. Data visualization: A practical introduction, 2018. Available from: https://books.google.com/books?hl=en&lr=&id=3XOYDwAAQBAJ&oi=fnd&pg=PR11&dq=R+language+data+visualization+and+exploration&ots=IZi47TcsUX&sig=alReIvMZsul1WjCAdOfy2t2iqgw [Accessed 28 Oct 2023].
28. Helkala K, *et al*. AI in Cyber Operations: Ethical and Legal Considerations for End-Users. Artificial Intelligence and Cybersecurity: Theory and Applications, 2022, 185-206.
29. Jáñez-Martino F, *et al*. A review of spam email detection: Analysis of spammer strategies and the dataset shift problem. Artificial Intelligence Review. 2023; 56(2):1145-1173.
30. Jawale D, Mahajan A, Shinkar K. Hybrid spam detection using machine learning. International Journal of Advance Research, Ideas and Innovations in Technology. 2018; 4(2):2828-2832. Available from: https://www.academia.edu/download/57544996/V4I2-1925.pdf [Accessed 28 Oct 2023].
31. Jovanović L, *et al*. Arithmetic optimization algorithm for spam detection. Portal.Sinteza.Singidunum.Ac.RsN Bačanin DžakulaSinteza 2022-International Scientific Conference on, 2022. portal.Sinteza.Singidunum.Ac.Rs. [online]. Available from: https://portal.sinteza.singidunum.ac.rs/paper/892 [Accessed 22 Sep 2023].
32. Kahimise J, Shava F. An analysis of social networking threats. In: International Conference on Cyber Warfare and Security. Academic Conferences International Limited, 2020. Available from: https://search.proquest.com/openview/d22f1d9dfe0fe7d0b1d1789d9c9b0010/1?pq-origsite=gscholar&cbl=396500 [Accessed 22 Sep 2023].
33. Kiilu KK. Sentiment Classification for Hate Tweet Detection in Kenya on Twitter Data Using Naïve Bayes Algorithm. [online], 2021. Available from: http://ir.jkuat.ac.ke/handle/123456789/5521 [Accessed 11 Oct 2023].
34. Kumar N, Sonowal S. Email spam detection using machine learning algorithms. In: 020 Second International Conference on Inventive Research in Computing Applications (ICIRCA). IEEE, 2020. Available from: https://ieeexplore.ieee.org/abstract/document/9183098/ [Accessed 23 Sep 2023].
35. Lang M, *et al*. mlr3: A modern object-oriented machine learning framework in R. Journal of Open Source Software. 2019; 4(44):p1903. Available from: https://joss.theoj.org/papers/10.21105/joss.01903.pdf [Accessed 28 Oct 2023].
36. Lee C, Kim H, Lee S. Analysis of Safety Considerations for Application of Artificial Intelligencein Marine Software Systems. Journal of Navigation and Port Research. 2022; 46(3):269-279. Available from: http://www.glonav.org/journal/view.php?doi=10.5394/KINPR.2022.46.3.269 [Accessed 12 Oct 2023].
37. Levy J, O'malley A. Don't dismiss logistic regression: The case for sensible extraction of interactions in the era of machine learning. BMC Medical Research Methodology. 2020; 20(1):1-15. Available from: https://bmcmedresmethodol.biomedcentral.com/articles/10.1186/s12874-020-01046-3 [Accessed 28 Oct 2023].
38. Liu H, *et al*. Poster: Video Chat Scam Detection Leveraging Screen Light Reflection. Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM, January 2019.
39. Lu H, Zhang X, Yang S. A learning-based iterative method for solving vehicle routing problems. In International Conference on Learning Representations. [online], 2019. Available from: https://openreview.net/forum?id=BJe1334YDH [Accessed 28 Oct 2023].
40. Mohammadzadeh H, Gharehchopogh FS. A novel hybrid whale optimization algorithm with flower pollination algorithm for feature selection: Case study Email spam detection. Computational Intelligence. 2021; 37(1):176-209.
41. Nandhini S, On JK. I.C. and 2020, undefined, n.d. Performance evaluation of machine learning algorithms for email spam detection, 2020. Ieeexplore.Ieee.Org. [online]. Available from: https://ieeexplore.ieee.org/abstract/document/9077835/ [Accessed 11 Oct 2023].
42. Nedaie A, Najafi A. Support vector machine with Dirichlet feature mapping. Neural Networks. 2018; 98:87-101. Available from: https://www.sciencedirect.com/science/article/pii/S0893608017302587 [Accessed 28 Oct 2023].
43. Nyamathulla S, *et al*. SMS Spam Detection with Deep Learning Model. 2022; 5:7006-7013, Journalppw.Com. Available from: https://www.journalppw.com/index.php/jpsp/article/view/8408 [Accessed 22 Sep 2023].
44. Pala Z, Atici R. Forecasting sunspot time series using deep learning methods. Solar Physics. 2019; 294(5):50. Available from: https://link.springer.com/article/10.1007/s11207-019-1434-6 [Accessed 28 Oct 2023].
45. Pisner D, Schnyer D. Support vector machine. Machine Learning, 2020, 101-121. Available from: https://www.sciencedirect.com/science/article/pii/B9780128157398000067 [Accessed 28 Oct 2023].
46. Sahoo SR, *et al*. Spammer Detection Approaches in Online Social Network (OSNs): A Survey. EAI/Springer Innovations in Communication and Computing, 2022, 159-180.
47. Siddiqui F, Suaib M. Machine Learning Approaches for Fake User and Spammer Detection: A Comprehensive Review and Future Perspectives. 2023; 13(3):23-46, Ijemr.Vandanapublications.Com, Available from: https://ijemr.vandanapublications.com/index.php/ijemr/article/view/1237 [Accessed 12 Oct 2023].
48. Subramanya R, Sierla S, Vyatkin V. From DevOps to MLOps: Overview and application to electricity market forecasting. Applied Sciences. 2022; 12(19):p9851. Available from: https://www.mdpi.com/2076-3417/12/19/9851 [Accessed 28 Oct 2023].
49. Sun N, *et al*. Near real-time twitter spam detection with machine learning techniques. International Journal of Computers and Applications, Taylor & Francis. 2020;

44(4), 338-348. Available from: https://www.tandfonline.com/doi/abs/10.1080/1206212X.2020.1751387 [Accessed 28 Oct 2023].

50. Tang S, *et al*. Clues in Tweets: Twitter-Guided Discovery and Analysis of SMS Spam. Proceedings of the ACM Conference on Computer and Communications Security, 2022, 2751-2764.
51. Theodorus A, *et al*. Short message service (sms) spam filtering using machine learning in Bahasa Indonesia. In: 2021 3rd East Indonesia Conference on Computer and Information Technology (EIConCIT). IEEE, 2021, 199-203. Available from: https://ieeexplore.ieee.org/abstract/document/9431859/ [Accessed 22 Sep 2023].
52. Thummadi B, Lyytinen K. How much method-in-use matters? A case study of agile and waterfall software projects and their design routine variation. Aisel.Aisnet.Org. 2020; 21(4):864-900. Available from: https://aisel.aisnet.org/jais/vol21/iss4/7/ [Accessed 28 Oct 2023].
53. Vejendla L, *et al*. Score based Support Vector Machine for Spam Mail Detection. In: 2023 7th International Conference on Trends in Electronics and Informatics (ICOEI). IEEE, 2023, 915-920. Available from: https://ieeexplore.ieee.org/abstract/document/10125718/ [Accessed 22 Sep 2023].
54. Verdugo J, Rodríguez M. Assessing data cybersecurity using ISO/IEC 25012. Software Quality Journal. 2020; 28(3):965-985.
55. Yurtseven İ, SB.-2021 6th I. and 2021, undefined, n.d. A review of spam detection in social media. Ieeexplore.Ieee.Orgİ Yurtseven, S Bagriyanik, S Ayvaz2021 6th International Conference on Computer Science and, 2021. ieeexplore.Ieee.Org. [online]. Available from: https://ieeexplore.ieee.org/abstract/document/9558993/ [Accessed 21 Sep 2023].
56. Zavrak S, Yilmaz S. Email spam detection using hierarchical attention hybrid deep learning method. Expert Systems with Applications. 2023; 233:p120997. Available from: https://www.sciencedirect.com/science/article/pii/S0957417423014793 [Accessed 23 Sep 2023].