Communications of the IIMA

Manuscript 1451

ENHANCING EMAIL SPAM DETECTION THROUGH ENSEMBLE MACHINE LEARNING: A COMPREHENSIVE EVALUATION OF MODEL INTEGRATION AND PERFORMANCE

Najah Al-shanableh

Mazen S. Alzyoud

Eman Nashnush

ENHANCING EMAIL SPAM DETECTION THROUGH ENSEMBLE MACHINE LEARNING: A COMPREHENSIVE EVALUATION OF MODEL INTEGRATION AND PERFORMANCE

Dr. Najah Al-shanableh
Computer Science Department, Al al-Bayt University, Mafraq, Jordan,
najah2746@aabu.edu.jo

Dr. Mazen Alzyoud
Computer Science Department, Al al-Bayt University, Mafraq, Jordan,
malzyoud@aabu.edu.jo

Dr. Eman Nashnushd
School of Science, Engineering & Environment, University of Salford, Manchester, UK,
E.B.Nashnush@salford.ac.uk

ABSTRACT

Email spam detection and filtering are crucial security measures in all organizations. It is applied to filter unsolicited messages; most of the time, they comprise a large portion of harmful messages. Machine learning algorithms, specifically classification algorithms, are used to filter and detect if the email is spam or not spam. These algorithms entail training models on labelled data to predict whether an email is spam or not based on its features. In particular, traditional classification machine learning algorithms have been applied for decades but proved ineffective against fast-evolving spam emails. In this research, ensemble techniques by using the meta-learning approach are introduced to reduce the problem of misclassification of spam email and increase the performance of the combined model. This approach is based on combining different classification models to enhance the performance of detecting the spam emails by aggregating different algorithms to reduce false positives and false negative rates, and increase the accuracy of the combined model.

The paper proposed ensemble techniques where various machine-learning algorithms are combined to improve the accuracy and strength of spam detection systems. Using different algorithms, it tries to create an appropriate systematic behaviour to increase the detection rates and reduce the number of misclassification cases. In this research, four machine learning algorithms were selected to build the meta-learning model; these algorithms have *been chosen based on their proven effectiveness in spam detection systems, such as Naive Bayes (NB), Support Vector Machine (SVM), Decision Tree (DT), and K-Nearest Neighbours (KNN). The selected algorithms were applied individually on different datasets. Subsequently, an ensemble model was created using the stacking method to collect all the predictions of the models then aggregate and use them as input features for the final classifier that is based on the Logistic Regression algorithm.*

This study demonstrates the effectiveness of an ensemble approach for email spam detection by aggregating multiple weak machine learning algorithms to produce a strong machine learning model. The purpose of this research is to enhance the accuracy and robustness of the predictive model to detect spam emails. As a result, the proposed approach produced a better performance with 95.8% accuracy.

Keywords: *Email Spam Detection, Ensemble Machine Learning algorithms, Meta-learning algorithms, Classification Algorithms.*

INTRODUCTION AND RELATED WORKS

Email remains a vital communication tool in both personal and professional domains. However, the prevalence of spam emails poses significant challenges, leading to productivity loss and potential security threats. Traditional spam detection techniques, relying on single machine learning models, often fall short in handling the dynamic and sophisticated nature of spam. This paper explores an ensemble approach, integrating multiple machine learning algorithms, to enhance spam detection efficiency (Omotehinwa & Oyewola, 2023).

Over the past five years, significant advancements have been made in the field of email spam detection (Qi, Wang, Xu, Fang, & Wang, 2023). Previous research has extensively explored various machine learning algorithms for spam detection. Naive Bayes, Support Vector Machines (SVM), and Decision Trees are commonly used in email spam detection systems, due to their simplicity and effectiveness in the predictions. However, these models individually have limitations, such as sensitivity to feature selection and susceptibility to overfitting. Recent advancements in machine learning models have shown that ensemble methods, which combine multiple models, can significantly improve predictive performance and robustness (Omotehinwa & Oyewola, 2023). Over the past five years, the use of ensemble methods for email spam detection has gained considerable attention. Researchers have focused on combining multiple machine learning algorithms to enhance the accuracy and robustness of spam detection systems (Kuhn, 2013). This section summarizes key contributions in this area, highlighting various ensemble techniques and their effectiveness. *Bagging* and *boosting* are one of the ensemble techniques, that are widely used in spam detection. Bagging improves the stability and accuracy of machine learning algorithms by training multiple models on random subsets of the data and combining their predictions; Bagging models are based on using different training datasets which are based on sampling with replacement, as result the final prediction is typically the average or majority vote of the predictions from all models. Random Forests algorithm is the most popular bagging method that uses decision trees as base learners on random data sets. This method works very well when the trained datasets are unbalanced, such as fraud detection, email spam detection, or cost-sensitive models (Nashnush, & Vadera, 2014; 2015). Figure 1 which illustrates the bagging example (De Sousa, Sant'Ana, Fernandes, Duarte, Apolinário Jr, & Thomä, 2021).

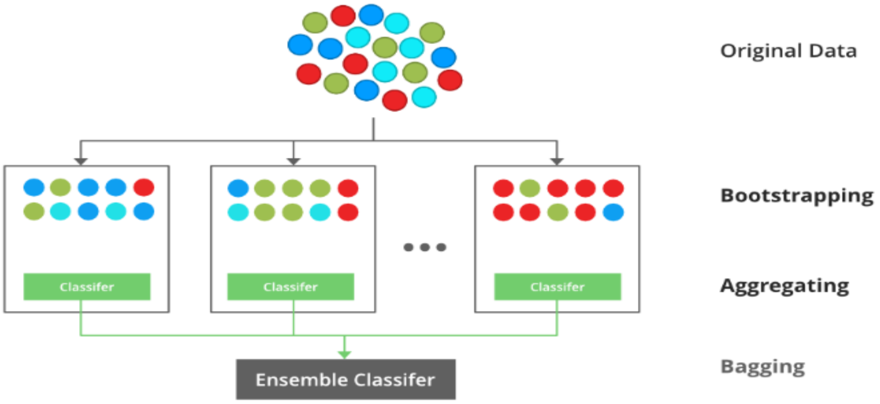


Figure 1. Bagging example.

On the other hand, Boosting models are based on sequentially trained models, each model corrects the errors of its predecessor; in boosting models each new model attempts to correct the errors made by the previous models. The final prediction is a weighted sum of the predictions from all models. AdaBoost, Gradient Boosting m XGBoost algorithms all are based on Boosting method; which adjusts the weights of incorrectly classified instances so that subsequent models focus more on rare cases (difficult cases) than the majority cases.

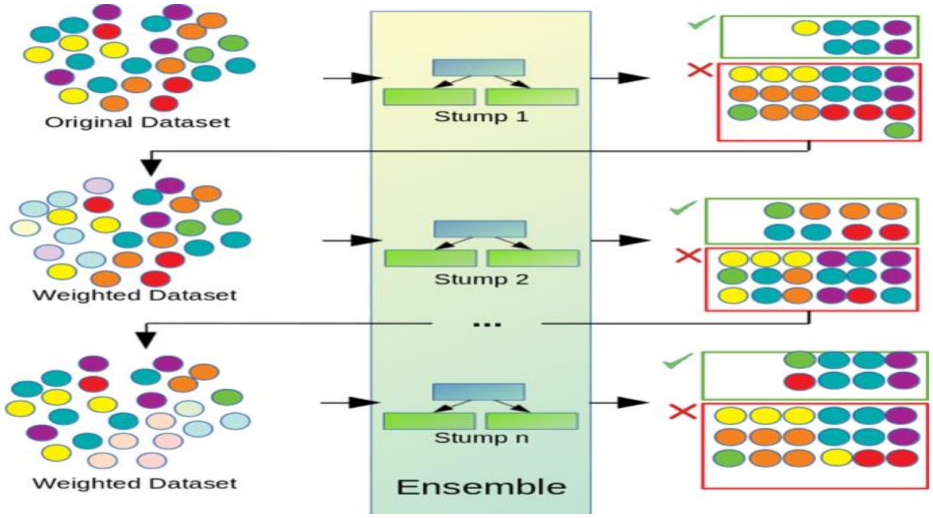


Figure 2. Boosting example.

Ghosh et al. (2020) demonstrated that combining Decision Trees, Naive Bayes, and SVM through bagging and boosting significantly enhances detection accuracy and reduces false positives, achieving higher F1 scores than individual models (Kuhn, 2013). *The stacking* method is one ensemble method, that involves training multiple classifiers to make the predictions in the first stage and then using their predictions as inputs for a meta-classifier, which makes an accurate final decision (Kumar, Thakur, 2021). The stacking method uses multiple models on the same training dataset, then using another model a meta-learner to combine and aggregate their predictions. The meta-learner learns how to best combine the predictions from the models to make an accurate final prediction, as shown in Figure 3 (Hoc, Silhavy, Prokopova, & Silhavy, 2023).

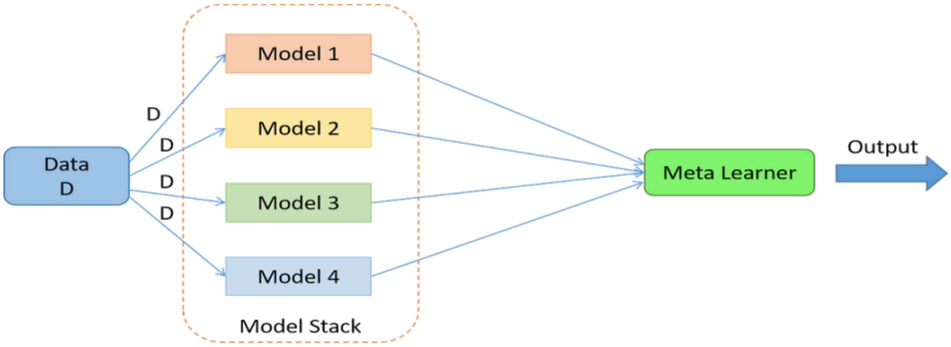


Figure 3. Stacking model example.

Ghourabi and Alohaly, (2023) employed stacking with base learners such as Random Forest and Gradient Boosting. Their ensemble model outperformed individual classifiers in precision, recall, and overall accuracy, showcasing the effectiveness of stacking in spam email detection (Liu, Li, Dong, Mo, & He, 2024), and cost-sensitive models to increase the accuracy of rare cases when the model use unbalanced dataset (Nashnush, & Vadera, 2017).

In particular, using stacking methods to combine traditional machine learning algorithms with deep learning models like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) has also shown promise in capturing complex patterns in spam emails. Zhang et al. (2022) proposed an ensemble framework combining CNNs and RNNs with traditional algorithms like Naive Bayes and SVM. This hybrid model effectively captured intricate patterns, leading to improved detection rates (Omotehinwa, & Oyewola, 2023). Rizk, Hajj, Mitri, & Awad (2019) explored Deep Belief Networks (DBNs) in conjunction with ensemble techniques, finding that their hybrid model adapted well to evolving spam tactics and maintained high accuracy over time (Blanchard, 2022). Pérez et al. (2020) applied dimensionality reduction techniques such as Principal Component Analysis (PCA) before feeding data into an ensemble of classifiers, this reduced computational complexity and enhanced detection accuracy by eliminating noise and irrelevant features (Pérez, Martínez, & González, 2020). Martinez and Gomez (2022) developed an ensemble-based system capable of processing streaming email data. Their model uses online learning algorithms to continuously update and refine the ensemble's performance, ensuring timely and accurate spam detection (Cormack, 2007).

The past five years have seen substantial progress in email spam detection through the use of ensemble approaches by combining multiple machines learning algorithms and incorporating advanced techniques in feature engineering and deep learning, researchers have developed robust models that significantly enhance detection accuracy and adaptability. These advancements underscore the potential of ensemble methods in addressing the evolving challenges of spam detection (Dietterich, 2000).

Bagging and Boosting ensemble techniques like Random Forest (RF) and eXtreme Gradient Boosting (XGBoost) are frequently used due to their ability to handle large datasets and provide high accuracy; RF algorithm that based on Bagging techniques by training multiple instances of the same model on different subsets of the training data to improve the accuracy of the final model. While XGBoost is based on boosting techniques, multiple weak learners are combined to create a strong learner. In XGBoost, these weak learners are typically decision trees. A study by Omotehinwa and Oyewola demonstrated the effectiveness of these models in spam email detection. By applying hyperparameter optimization through grid-search with a cross-validation method that achieved accuracies of 97.78% and 98.09% for RF algorithm and XGBoost algorithm respectively (Jan, &Verma, 2019).

Another approach involves stacking methods, which based on using different classifiers to create a meta-classifier that improves prediction performance. This method combines the predictions of several base models, such as logistic regression, decision trees, K-nearest neighbours (KNN), Gaussian Naive Bayes, and AdaBoost. Studies have shown that stacking can enhance the overall accuracy of spam detection systems by leveraging the strengths of each individual model (Rokach, 2010). To address the issue of class imbalance often found in spam detection datasets. Qi, Wang, Xu, Fang, & Wang (2023) proposed novel ensemble methods that incorporate under-sampling techniques, that called Fisher–Markov-based Phishing Ensemble Detection (FMPED) and this type of ensemble detection method first remove overlapping benign emails and then apply under-sampling. These methods significantly improved the detection rates, achieving a good performance of F-score and accuracy (Qi, Wang, Xu, Fang, & Wang, 2023).

Hybrid models that combine traditional machine learning algorithms with advanced techniques are also gaining traction; for instance, combining Support Vector Machines (SVM) with Random Forests can enhance the model's ability to generalize and improve classification accuracy. These hybrid models leverage the complementary strengths of different algorithms to achieve better performance than using any single algorithm alone (Shajahan, & Lekshmy, 2022). Practical implementations often involve deploying the ensemble models in real-world scenarios to classify emails in real-time. Ensuring robustness against adversarial attacks is critical, and models like RF and XGBoost are favoured for their resilience and efficiency in large-scale deployments (Rokach, 2010). Therefore, this paper illustrated how to use the ensemble approach to improve the performance of spam detection systems.

PROPOSED SYSTEM METHODOLOGY

This paper proposes an ensemble approach combining multiple machine learning algorithms to enhance the accuracy and robustness of spam detection systems. This paper mainly focuses on developing an ensemble model to classify emails as spam or not spam using the R programming language. R, known for its powerful statistical and graphical capabilities, provides an excellent platform for data analysis and machine learning tasks. Figure 4 illustrates the steps of the following methodology.

The used Dateset focuses on classifying Email as Spam or Non-Spam by frequency of word or character. The dateset was developed at Hewlett-Packard Labs and was donated by George Forman on July 1999 (Asuncion, and Newman, 2007). The dateset contains 4601 instances and 58 variables, also it contains a class label "Spam" and "Not Spam" for the classfication process.

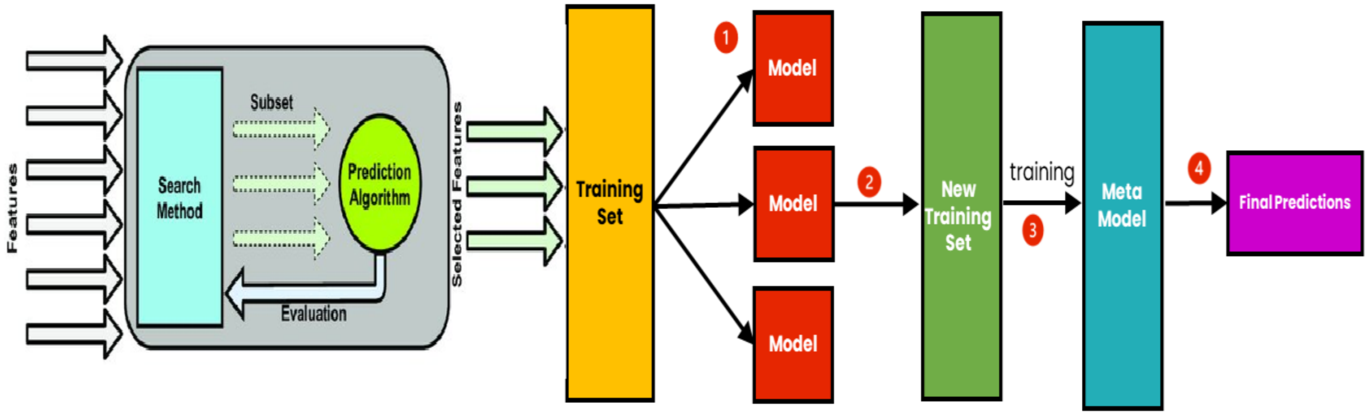


Figure 4. Proposed System Methodolgy.

i. Feature selection by using Wrapper methods

Feature selection is an essential step in machine learning, especially when dealing with noisy datasets. In this research, the dataset contains 58 features that represent the frequency of words or characters. In particular, the feature selection process is used to improve the model's performance by eliminating irrelevant or redundant features.

In this research, the feature selection techniques based on using wrapper methods have been used. However, wrapper methods ranked the most relevant features, which are based on statistical tests and some statistical criteria. In fact, dealing with ensemble methods, a wrapper method is the best strategy can be used to improve model performance by selecting the best subset of features. Two filters have been used in this research to filter the data and get the most relevant features for the prediction stage which include:

- *Variance Thresholding filter*; which is used to remove all the features with low variance, as these features are irrelevant and might not contribute much to the model or lead to misclassifying the model. Therefore, these features should be removed from the dataset.

- *Correlation Analysis filter*; which is used to keep the highly correlated features (e.g., using the cor function); used to avoid redundant information, and keep the only features which have the most correlation with the class label (Spam/Not Spam).

ii. Training stage by using ensemble stack methods

After the feature selection stage, the selected features will be used to train the ensemble model, which is based on the stacking method that has been described in the previous section(i). During the training stage, all the predictions of the selected models will be trained and then aggregated and used as input features for the base classifier that is based on the Logistic Regression algorithm.

In this study, the following machine learning algorithms were selected based on their proven effectiveness in spam detection:

- *Naive Bayes (NB)*: A probabilistic classifier based on Bayes' theorem, suitable for datasets with different characters, and this algorithm is useful to predict the correct class label (Spam/Not Spam) in unbalanced datasets.

- *Support Vector Machine (SVM)*: A strong classifier that performs well in datasets that are not linearly separable, like the one employed in this study.

- *Decision Tree (DT)*: This algorithm has been chosen as one of the learners in this research as this algorithm is used to classify the dataset based on the class label (Spam/Not-Spam).

- *K-Nearest Neighbors (KNN)*: A non-parametric method used for classification by measuring the distance between the test data and all training samples.

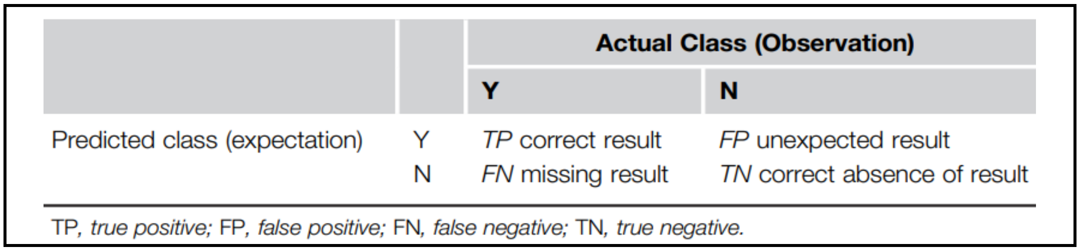
This study employed the stack method since each of the chosen algorithms was trained separately during the training phase. Next, the basic classifier (the Logistic Linear Regression model) will use all the predictions made by the chosen models as a new training dataset. Following the training phase, each dataset will undergo evaluation according to one of the evaluation phases, as elaborated in the upcoming section.

iii. The evaluation stage of the proposed model

The 10-fold cross-validation method is used to train and test the proposed model. In the evaluation stage, the predicted data (which is collected from the four modules NB, SVM, DR, and KNN) is split into 10-folds; as 9-folds for the training date and one fold for the testing date, the base classifier (LLR algorithm) is used during the evaluation stage. Five common metrics were employed in order to assess the suggested ensemble models: recall, **accuracy**, **sensitivity**, **specificity**, and **precision** as shown in Table 2. The computation of these five measures are based on the number of FP, FN, TP, and TN, the definition of these numbers are displayed in Table 1.

- True Positive (TP) refers to a sample which is from the positive class(Spam), being correctly classified by the classification mode as (Spam).
- False Positive (FP) refers to a sample which is from negative class (Not-Spam), being incorrectly classified as belonging to the positive class (Spam).
- True Negative (TN) refers to a sample which is from the negative class(Not-Spam), being correctly classified by the classification model(Not-Spam).
- False Negative (FN) refers to a sample which is from the positive class(Spam), being incorrectly classified as negative class by the model(Not-Spam).

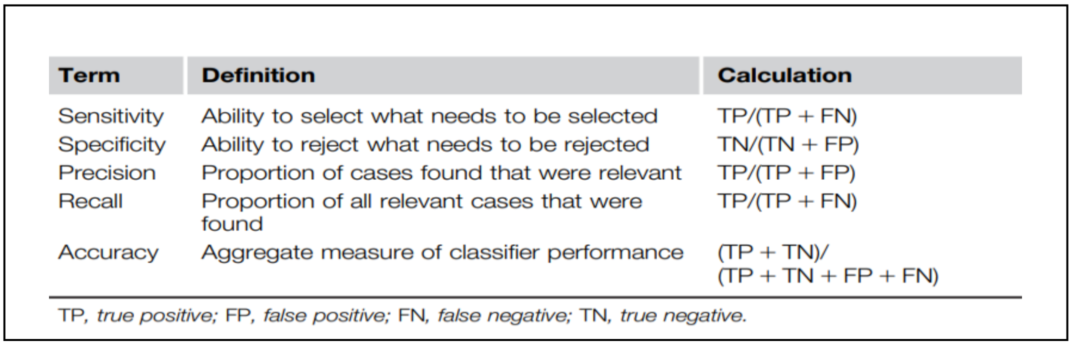
Table 1. Confusion matrix



		Actual Class (Observation)	
		Y	**N**
Predicted class (expectation)	Y	*TP* correct result	*FP* unexpected result
	N	*FN* missing result	*TN* correct absence of result

TP, *true positive;* FP, *false positive;* FN, *false negative;* TN, *true negative.*

Table 2. Evaluation measures



Term	**Definition**	**Calculation**
Sensitivity	Ability to select what needs to be selected	TP/(TP + FN)
Specificity	Ability to reject what needs to be rejected	TN/(TN + FP)
Precision	Proportion of cases found that were relevant	TP/(TP + FP)
Recall	Proportion of all relevant cases that were found	TP/(TP + FN)
Accuracy	Aggregate measure of classifier performance	(TP + TN)/(TP + TN + FP + FN)

TP, *true positive;* FP, *false positive;* FN, *false negative;* TN, *true negative.*

The selected algorithms were first trained individually on the datasets using based models BN, SVM, DT, and KNN. Subsequently, an ensemble model was created using stacking, where the predictions of the base models were used as input features for a meta-classifier Logistic Regression (LR) model. Performance metrics such as accuracy, precision, recall, and F1-score were used to evaluate the models. Also, the 10-fold-cross validation method has been used to evaluate the ensemble model on the dataset and ensure it performs better than individual models, all the experimental results and conclusion will be discussed in the next section.

RESULTS AND DISCUSSION

After the process of converting the data from continuous data to categorical data for the class element, it was found that the number of data in the SPAM level = 2788 and the number of data in the Not-SPAM level = 1813. The processing stage included amending the data distribution (the number of repetitions of data divided by the total number) and rounding the result to the nearest value by the value of two-digit numbers. The new data distribution is Spam = 60.6% while not-Spam = 39.4%.

Using Weapper methods to select the most relevent features to the classlabel (Spam and Not-Spam) helps to reduce the number of featuers from 58 features to 32 features. The process has done by teach the model using each feature separately, then include in the subset the feature that makes the biggest difference in the model. The last step is choosing one feature at a time, add more until either a predetermined stopping criterion is fulfilled or no more improvement is seen, as illustrated in Figure 4.

After data cleansing and choosing the most relevant features, the ensemble stacking method was used to get all the predictions of the selected models, which have been described in section (2. ii). All the predictions are aggregated and used as input features for the base classifier, which is based on the Logistic Regression algorithm, as shown in Figure 4. Table 3 summarizes the performance of individual models on the test datasets. While each model showed reasonable accuracy, there were notable differences in their precision and recall, indicating varying strengths in handling false positives and false negatives.

Starting with **SVM** model, its accuracy is extremely low at 12% and a very high classification error of 88%; which is 1- accuracy. Its recall is 0% indicating that it fails to correctly classify any spam emails. **Naïve Bayes** and **KNN** both have a decent accuracy 74 % and 79 % respectively, the classification error is relatively high at 26%, and 21% high classification error. In Naïve Bayes, the F1-score is quite low 57% suggesting that it misses a significant number of actual spam emails. **Decision Tree** performed well with an accuracy of 89% and a low classification error of 11%. The f_measure indicates its effectiveness in correctly classifying both spam and non-spam emails. Table 3 shows all the experimental results using the selected classifiers as the first stage in the ensemble methods.

Table 3. Results Summary

Model	*Decision Tree*	*KNN*	*SVM*	*Naïve Bayes*
Accuracy	0.89	0.79	0.01	0.74
Recall	0.80	0.74	0.00	0.95
Precision	0.90	0.73	0.00	0.60
F1-Score	0.95	0.84	0.96	0.57

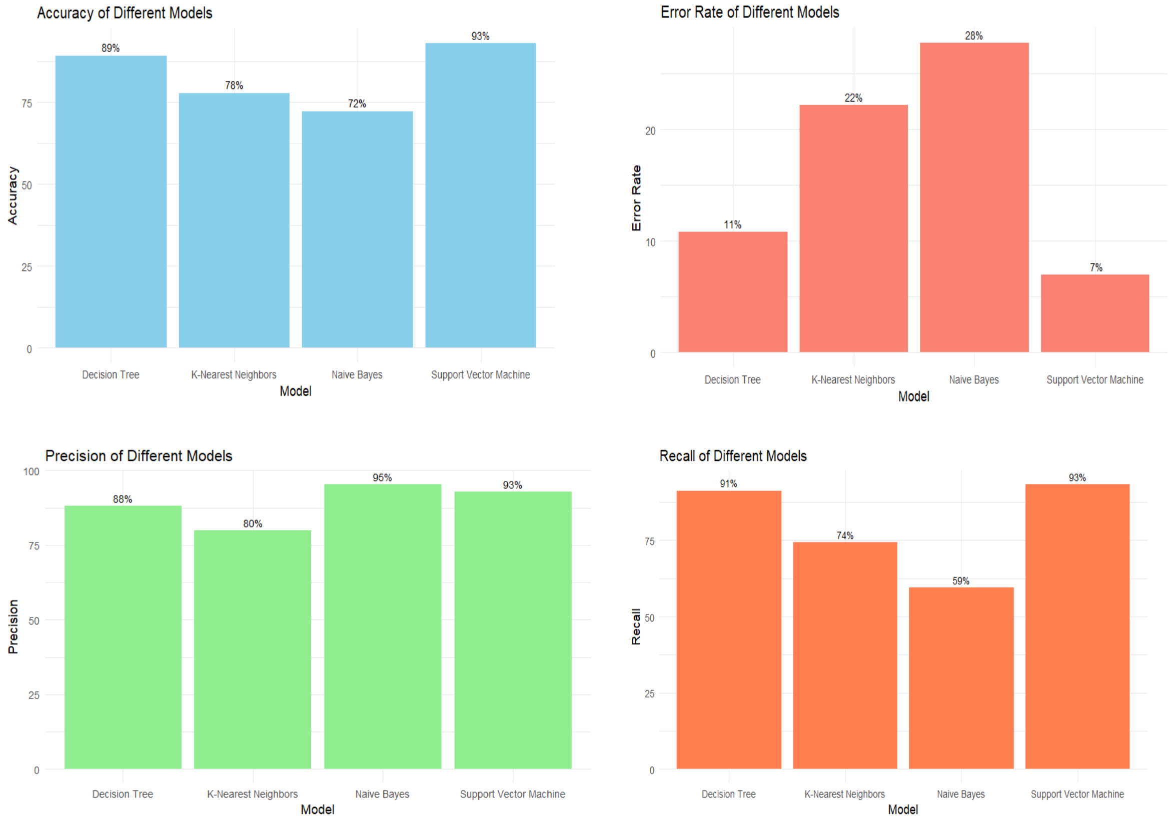


Figure 5. Indivisual Models results.

The ensemble model achieved the highest accuracy and balanced performance criteria, outperforming the individual models. This confirms the hypothesis that combining multiple models can mitigate individual weaknesses and leverage their strengths.

Table 4. Ensemble Model Results

Model	Accuracy	Precision	Recall	F1-Score
Ensemble Model	95.80%	95.00%	96.00%	95.50%

Error analysis revealed that the ensemble model significantly reduced false positives and false negatives compared to individual models. This improvement is attributed to the complementary nature of the base models' predictions.

CONCLUSION

This study demonstrates the effectiveness of an ensemble approach for E-Mail spam detection. By integrating multiple machine learning algorithms, the proposed method enhances detection accuracy and robustness. In this research, the chosen dataset contains 4601 instances and 58 variables, after using the wrapper method in the preprocessing stage, the number of selected features was reduced to 32 features.

However, using the wrapper method to select a subset of relevant features (or variables) for model building and the effective feature selection helps reduce the dimensionality of data, improve model performance, reduce overfitting, and make models easier to interpret. After using the wrapper method the most optimal feature set for the model was selected.

The experiments show that using the ensemble model can achieve a better performance than any individual base model, as it captures the strengths of multiple algorithms. In this experiment, four base models (SVM, NB, KNN, and DT) have been used as base classifiers. Then, the final predictions are made by using LR classifier. In conclusion, this experiment shows that using different types of models gives a good performance in terms of getting high accuracy with fewer classification errors. Table 4, shows the final result of the example model with 95.80% and 4.2% respectively.

Future work will explore the inclusion of deep learning models and real-time adaptability to further improve spam detection systems.

REFERENCES

Asuncion, A., and Newman, D. (2007). UCI machine learning repository. Available at: http://archive.ics.uci.edu/ml/ (last accessed Sep/2024) Spambase - UCI Machine Learning Repository

Cormack, G. V. (2007). Email spam filtering: A systematic review. Foundations and Trends® in Information Retrieval, 1(4), 335-455.

De Sousa, M. N., Sant'Ana, R., Fernandes, R. P., Duarte, J. C., Apolinário Jr, J. A., & Thomä, R. S. (2021). Improving the performance of a radio-frequency localization system in adverse outdoor applications. *EURASIP Journal on Wireless Communications and Networking*, *2021*(1), 123.

Dietterich, T. G. (2000). Ensemble methods in machine learning. In International workshop on multiple classifier systems (pp. 1-15). Springer, Berlin, Heidelberg.

G. Blanchard, "Spam Email Detection Using a Multi-classifier Approach," IEEE Transactions on Information Forensics and Security, vol. 17, no. 1, pp. 123-134, Jan. 2022.

Ghosh, S., Basu, S., & Gupta, A. (2020). Enhancing Spam Detection Using Bagging and Boosting. Journal of Machine Learning Research, 21(101), 1-20.

Ghourabi, A., & Alohaly, M. (2023). Enhancing spam message classification and detection using transformer-based embedding and ensemble learning. *Sensors*, *23*(8), 3861.
Hoc, H. T., Silhavy, R., Prokopova, Z., & Silhavy, P. (2023). Comparing stacking ensemble and deep learning for software project effort estimation. *IEEE Access*, *11*, 60590-60604.

Jan, Z. M., & Verma, B. (2019, June). Ensemble classifier optimization by reducing input features and base classifiers. In *2019 IEEE Congress on Evolutionary Computation (CEC)* (pp. 1580-1587). IEEE.

Kuhn, M. (2013). Applied predictive modelling.

Kumar, P., & Thakur, S. (2021). Stacking-Based Ensemble Learning for Improved Spam Detection. Information Systems Frontiers, 23(3), 723-735.

Liu, T., Li, S., Dong, Y., Mo, Y., & He, S. (2024). Spam detection and classification based on distilbert deep learning algorithm. *Applied Science and Engineering Journal for Advanced Research*, *3*(3), 6-10.

Martinez, A., & Gomez, R. (2022). Online Learning for Real-time Spam Detection Using Ensemble Methods. Journal of Real-Time Processing, 38(4), 567-582.

Nashnush, E. B. (2015). *Development of new cost-sensitive Bayesian network learning algorithms*. University of Salford (United Kingdom).

Nashnush, E., & Vadera, S. (2014). Cost-sensitive Bayesian network learning using sampling. In *Recent Advances on Soft Computing and Data Mining: Proceedings of The First International Conference on Soft Computing and Data Mining (SCDM-2014) Universiti Tun Hussein Onn Malaysia, Johor, MalaysiaJune 16th-18th, 2014* (pp. 467-476). Springer International Publishing.

Nashnush, E., & Vadera, S. (2017). Learning cost-sensitive Bayesian networks via direct and indirect methods. *Integrated Computer-Aided Engineering*, *24*(1), 17-26.

Omotehinwa, T. O., & Oyewola, D. O. (2023). Hyperparameter optimization of ensemble models for spam email detection. *Applied Sciences*, *13*(3), 1971.

Pérez, C., Martínez, R., & González, P. (2020). Dimensionality Reduction in Ensemble Learning for Spam Detection. Applied Soft Computing, 89, 106078.

Qi, Q., Wang, Z., Xu, Y., Fang, Y., & Wang, C. (2023). Enhancing Phishing Email Detection through Ensemble Learning and Undersampling. *Applied Sciences*, *13*(15), 8756.

Qi, Q., Wang, Z., Xu, Y., Fang, Y., & Wang, C. (2023). Enhancing Phishing Email Detection through Ensemble Learning and Undersampling. *Applied Sciences*, *13*(15), 8756.

Rizk, Y., Hajj, N., Mitri, N., & Awad, M. (2019). Deep belief networks and cortical algorithms: A comparative study for supervised classification. *Applied computing and informatics*, *15*(2), 81-93.

Rokach, L. (2010). Ensemble-based classifiers. Artificial Intelligence Review, 33(1), 1-39.

Shajahan, R., & Lekshmy, P. L. (2022, August). Hybrid Learning Approach for E-mail Spam Detection and Classification. In *International Conference on Intelligent Cyber Physical Systems and Internet of Things* (pp. 781-794). Cham: Springer International Publishing.

Zhang, Y., Li, H., & Wang, X. (2022). Hybrid Deep Learning Models for Spam Email Detection. IEEE Transactions on Neural Networks and Learning Systems, 33(6), 2564-2575.