International Journal of Computational and Experimental Science and ENgineering (IJCESEN)

Vol. 11-No.3 (2025) pp. 7066-7074
http://www.ijcesen.com

ISSN: 2149-9144

Copyright © IJCESEN

Research Article

An Intelligent Ensemble Machine Learning Framework for Enhanced Email Spam Detection Under Class Imbalance

Garima Mishra[1]*, Parth Gautam[2]

[1] Department of Computer Science and Application ,Mandsaur University Mandsaur, Madhya Pradesh, India
* **Corresponding Author Email:** mishragarima2708@gmail.com- **ORCID:** 0000-0002-5247-2350

[2]Department of Computer Science and Application ,Mandsaur University Mandsaur, Madhya Pradesh, India
Email: parth.gautam@meu.edu.in - **ORCID:** 0000-0002-5247-0050

Article Info:

DOI: 10.22399/ijcesen.5537
Received : 22 August 2025
Revised : 15 October 2025
Accepted : 20 October 2025

Keywords

Cybersecurity,
Email Spam,
Class Imbalance,
Ensemble Machine Learning,
Voting classifier.

Abstract:

Email is among the most used and effective internet communication and data or message sharing methods. With the importance and high usage of emails, spam mail has also grown at a great rate. Email spam detection is an essential task for maintaining secure and efficient email communication by distinguishing unwanted messages from legitimate emails. This research presents a methodology for accurate Spam email categorization using Spam Base-UCI dataset based on ML. The framework applies data preprocessing, Chi-Squared-based Select Best feature selection to retain the 35 most informative features, StandardAero for feature normalization, and SMOTE-ENN to address class imbalance. Three classification models, namely MLP, Extra Trees, and LightGBM, are developed and integrated using a Soft Voting Classifier. The models are evaluated using accuracy, precision, recall, F1score, sensitivity, specificity, ROC-AUC, and Precision-Recall analysis. Experimental outcomes demonstrate that the Voting Classifier achieves the best performance, with 99.02% accuracy, 99.43% precision, 98.58% recall, and 99.00% F1score, outperforming the individual classifiers and several existing approaches. The results demonstrate that integrating feature selection, class balancing, and complementary ensemble models provides an effective and robust approach for distinguishing Spam Emails and Legitimate Emails. The values of the research performance measures indicate that ML models have the potential to enhance the adoption of cybersecurity solutions in combating email spam attacks.

1. Introduction

The internet's technological innovations have completely changed our world by making it possible for knowledge to be shared instantly around the world. But there is a serious downside to this advancement: the increase in spam in our inboxes. Email spam is a widespread problem that affects consumers worldwide and includes a wide range of unsolicited messages. The proliferation of spam is a sad side effect of the convenience that email has brought to communication. It's now essential to effectively separate spam from valid emails. Unsolicited commercial email (UCE), one of the most well-known types of spam, bombards receivers with unsolicited sales spiels for goods or services they never agreed to receive [1]. Furthermore, emails are dangerous because they might fool receivers into divulging private information by imitating official correspondence from reliable sources. Emails have made it easier for people to work together because they are a quick and affordable way to communicate [2]. They have greatly expedited communication and information exchange on personal as well as professional grounds. However, because of escalating usage and reliance on emails, consumers are now more susceptible to cybersecurity risks encompassing malware infections, spam attacks, and sundry types of wringing. Researchers have looked into a number of ways to ameliorate spam detection's efficacy to solve the difficulties that come with email communication ML algorithms have been widely used in a variety of email spam during the past ten years due to significant advancements in the field of AI. Among commonly used ML techniques [3] for spam classification are supervised learning methods, such as Random Forest, SVMs, XGBoost, NB, and

ANNs [4][5]. With their potentiality to grasp data patterns, attributes, and email structures, these incredibly adaptable algorithms offer encouraging results when it comes to spotting phishing emails. Additionally, by including balance and selection procedures that are normally carried out manually by human experts, machine learning approaches may effectively classify email spam [6]. Building email spam detection models using ML approaches is the driving force behind this project, which aims to accurately identify spam emails from legitimate emails. The following research contributions of this work are:

- To develop an integrated ML framework for effective email spam detection using pre-processing, feature selection, class balancing, and ensemble learning.
- To identify the most informative email characteristics using Chi-Squared Select Best feature selection and reduce the dimensionality of the input data.
- To address class imbalance using SMOTE-ENN by combining minority-class oversampling with noise and ambiguity removal.
- To develop and comparatively evaluate MLP, Extra Trees, and LightGBM models for spam and legitimate email classification using accuracy, confusion matrix, ROC-AUC, and Precision-Recall analysis.
- To improve classification robustness by integrating the base models through a Soft Voting Ensemble that combines their complementary prediction probabilities.

A. Organization of the paper

The remnant article is arranged as follows: The relevant works for email spam detection are covered in Section II. Using flowcharts, and models, Section III delineates the proposed process. Section IV delineates experimental results and a discussion of comparison. Section V concludes by outlining the limits and discussion of future work.

2. Literature Reviews

An overview of relevant work is given in this section. Sundry studies have scrutinized email spam detection using both ML and DL techniques. Research has progressed over time from conventional classifiers to ones that target various spam identification issues. According to the literature, A. M. John and S. S. Joseph (2026), introduces computationally efficient and interpretable ML framework for spam detection in SMS and email messages. The TF-IDF technique and classifiers including LR, RF and Multinomial NB are integrated through soft-voting ensemble strategy. Experimental evaluation shows that the system achieves an accuracy of 97.7% for SMS spam classification [7]. Another work by T. Chanda et al. (2026), proposed model is developed by combining the strengths of SVM with DNN & CNN to achieve a recall value of 97.41% [8]. Furthermore, V. Sowmitha et al. (2025), user's device locally without exchanging data since the solution uses federated learning with LSTM. The accuracy rate of Federated Learning (FL) is 94.5%, which is higher than that of Centralized Learning (CL), which is 91.0%. Email spam detection using federated learning and LSTM is quick and safe [9]. Additionally, L. Zou (2024) analyzes spam email detection using two different NLP models: one that relies on counts and another that relies on inferences. The study makes use of a real-life email dataset and focuses on two models: TFIDF-SVD-RF and Word2Vec-BiGRU-NN. The inference-based model achieved a 98.39% accuracy rate and a 93.92% F1 score [10]. After this, N. RamojiRao et al. (2024) included testing how well SVM and a new NB Classifier performed. The experiment's sample size was set at 25 emails per group based on an 80% power G-power analysis. The findings demonstrated that the innovative NBC attained a 94% accuracy rate, surpassing SVM's 89.78% accuracy rate and comparatively low mean error [11].Alongside, A. Saini et al. (2023) article helps with the creation of a spam detection model that uses several categorization approaches to combat spam email. The NB, K*, J48, and RF classification algorithms are all used in this model. The prediction classifier that has exhibited the best accuracy of 95.48% is the random forest model [12]. Lastly, V. I. Del Rosario et al. (2023), research aims to develop a self-contained system that can distinguish between ham and spam using Distill BERT. All all, 11,214 legitimate emails and 7,287 spam emails made it into the model's training set. The gadget operates on a Raspberry Pi's Flask server. 95spam and 97 ham emails were successfully categorized out of 100spam and 100 ham emails. The gadget has an MCC of 92.02% and an accuracy of 95.99%. Positive correlations and predictions were obtained [13].The reviewed studies demonstrate the increasing use of ML, DL, NLP, federated learning, and ensemble techniques for spam detection, with reported performance varying across datasets and approaches. Yet, there are also a number of limitations such as lack of deep integration of feature selection and class-imbalance handling, inadequate comparison of various ML classifiers, and the use of few ensemble methods to boost robustness. Existing studies are also largely data-specific or communication platform dependent, and

may limit generalizability. Hence, an integrated and systematically evaluated framework is required to integrate informative feature selection, multiple complementary classifiers, and soft voting ensemble learning for reliable email spam detection.

3. Methodology

This study aims to present a ML based framework for the effective detection of email spam in the Spam Base-UCI dataset. The methodology comprises data pre-processing, selection of 35 most significant features, standard scaling and SMOTEENN for class imbalance. The data are split into 70% training sets and 30% Testing Sets. The accuracy, precision, recall, sensitivity, specificity and F1score metrics are utilized to develop and evaluate the models: MLP, Extra Trees, LightGBM and Voting Classifier. Fig 1 shows the model that tries to correctly classify spam and actual email.

B. Data Collection

In this research, a Spam base [14]database attained from the UCI ML Repository and generated by George Forman, Erik Reeber, Mark Hopkins, and Jaap Suermondt was selected. The dataset consisted of 4,601 email messages, each having 195 samples, and a 57-feature set, including one particular feature which indicated a class. The dataset contains email messages that are both spam and non-spam from their postmaster and those who reported spam. Furthermore, non-spam messages were collected from personal emails, personal letters and finished work. The data set consisted of selected emails appropriate for the evaluation of spam filtering systems. The majority of characteristics were the frequency of matching letters and words in the e-mail messages.

C. Data pre-processing

A systematic data pre-processing method is followed prior to feature selection and model development. The data is scanned for missing or null data, duplicate data, inconsistent data, and inappropriate data types. Duplicate and invalid observations are deleted to ensure data quality if they are found. The target variable is checked to make sure that both spam and non-spam classes are properly balanced with binary labels. The feature columns are converted to appropriate numeric types as required. In addition, the input variables and output variable are purged to avoid bias from the output variable affecting the input variable transformation. The dataset is prepared for feature selection and ML operations after this pre-treatment step makes sure it is clean and consistent.

D. Feature and Target Separation

After pre-processing, a dataset is split into two parts, the input Feature Matrix (X) and the Target Vector (Y). The Feature Matrix consists of the numerical characteristics of email messages, while the target variable is used to determine which class an email belongs to (spam or legitimate). This separation defines the structure for the prediction during the feature selection and classification processes.

E. Feature Selection Using Chi-Squared Test

The Chi-Squared (χ^2) test with Select Best is employed to identify the most informative features associated with the target class. Each feature is statistically tested with regard to the spam and non-spam classes. The top 35 features with the highest Chi-Squared scores are selected, as shown in Fig 2. It is a process that compresses the data, removes attributes that contribute less to the discriminatory information for spam classification, lowers the dimension of the data, and preserves attributes that contain more discriminatory information.

F. Standard Scaling

The chosen 35 features have been standardized using the StandardScaler so that all the input variables are on a similar numerical scale. Standardization equals measuring each feature around an average of zero and a standard deviation of one. The transformation is mathematically expressed as Equation .(1):

$$Z = \frac{X-\mu}{\sigma} \quad \square\square\square$$

where X represents the initial feature value, μ is the mean of the feature, and σ stands for its standard deviation. This makes models like MLP more stable during learning by reducing the impact of changing feature ranges.

G. Data Balancing

The SMOTE-ENN approach is used to correct the imbalance among spam and legal email classes. SMOTE creates fresh SyntheticSamples for the MinorityClass rather than merely repeating previous observations. A SyntheticSample is generated using Equation .(2):

$$x_{new} = x_i + \lambda(x_{nn} - x_i), \quad \lambda \in [0,1] \quad \square\square\square$$

where x_i represents a Minority Class sample, x_{nn} denotes one of its nearest Minority Class neighbors, and λ is a Randomly Selected value between 0 and 1. After oversampling, ENN is applied to remove noisy or ambiguous samples. For a sample x_i, its class can be retained when it agrees with the Majority Class among its KNN as in Equation.(3)

$$\hat{y}_i = mode\{y_1, y_2, \ldots . y_k\} \quad \square\square\square$$

Samples whose actual class differs from the neighbourhood majority are removed. Table I illustrates the class distribution before and after applying SMOTE-ENN. Therefore, SMOTE-ENN simultaneously balances the class distribution and cleans overlapping or noisy observations, enabling the subsequent classifiers to learn more reliable decision boundaries for spam and legitimate email classification.

Table 1. *Class distribution after before smote-enn*

Stage	Class 0	Class 1	Total
Before SMOTE-ENN	2,788	1,812	4,600
After SMOTE-ENN	2,345	2,414	4,759

H. Training and Testing Data Split

The train_test_split technique is used to split the resampled SMOTE-ENN dataset into two parts: one for training and one for testing. A 70:30 ratio is adopted, where 70% of the data is utilized for model training and 30% for independent testing.2 The random_state=17 ensures reproducibility of the data partitioning across experiments.

İ. Proposed Models

Three separate ML models—MLP, ExtraTreesClassifier, and LightGBM—are developed to assess the effectiveness of a proposed framework for email spam detection. These predictions are then combined with a Soft Voting Ensemble Classifier to get a better and more reliable spam classification result.

1) Multi-Layer Perceptron (MLP) Model

An MLP classifier is used to detect the nonlinear relationships in the selected email features. The model has a single hidden layer of 53 neurons (hidden_layer_sizes=(53,)). The nonlinear learning capability is added using the ReLU activation function. The number of training samples to process at each weight update is controlled by the batch size (batch_size=29), and the initial learning rate is 0.0003 (learning_rate_init=0.0003). The maximum number of iterations is 1907 (max_iter=1907). By repeatedly adjusting the weights of the network to reduce the classification loss, the MLP learns the classification patterns.

2) Extra Trees Classifier

The Extra Trees Classifier is used as a random forest or an ensemble of randomized decision trees. It uses 73 trees (n_estimators=73) to create several decision boundaries, enhancing the robustness of the predictions. Log-loss is used as the quality function for candidate splits (criterion='log_loss'). A minimum number of samples (min_samples_split=4) required to split an internal node is set to 4, which helps prevent overly complicated splits, thereby limiting the growth of the tree. The random state is set to 53 (random_state=53) to make sure that the results are reproducible. Randomized trees' predictions can be combined to capture nonlinear relations and feature interactions.

3) LightGBM Model

The LightGBM classifier is used to efficiently learn complex relationships in the selected email features. The model applies 360 boosting iterations (n_estimators=360) which means that multiple sequential trees are built to progressively minimize classification errors. The learning rate will be 0.057 (learning_rate=0.057), this is the value of each single tree in the final model. The parameter num_leaves=140 specifies the maximum number of leaves in each tree and is enough for learning complicated decision boundaries. The random state is fixed at 17 (random_state=17) for repeatability of experiments.

4) Soft Voting Classifier

Finally a SoftVotingClassifier is utilized to combine the predictions of the models such as MLP, Extra Trees, and LightGBM. The three models trained are base estimators while the soft voting approach fuses class probabilities instead of class labels. The final prediction is made from the three individual probabilities which are aggregated to take advantage of the complementary strength of the three classifiers and to give a more robust spam/non-spam classification.

J. Performance Matrix

The proposed models are evaluated using a variety of classification metrics, including recall, accuracy, precision, sensitivity, specificity, ROC-AUC, and Precision-Recall (PR) calculations. The total percentage of properly classified emails is measured by accuracy (ACC), while the fraction of correctly detected SpamEmails among all emails projected as spam is shown by precision (PRE). Recall (REC) or sensitivity (SEN) measures the model's ability to correctly detect ActualSpamEmails, whereas specificity evaluates its ability to correctly identify legitimate emails. The F1-score (F1) is a balanced measure of precision and recall that is especially helpful when the ClassDistributions are uneven. The specificity (SPE) of an email detection model is defined as the percentage of accurate TN predictions relative to the total number of valid emails. The measures evaluate as Equation.(4 to 7):

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \quad (4)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (5)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \quad (6)$$

$$F1 - Score = \frac{2*Precision*Recall}{(Precision+Recall)} \quad (7)$$

Where: TP (True Positive) denotes correctly detected spam, TN (True Negative) denotes correctly detected legitimate emails, FP (False Positive) denotes legitimate emails incorrectly classified as spam, and FN (False Negative) denotes spam emails incorrectly classified as legitimate. In addition, the confusion matrix is used to analyze TP, TN, FP, and FN predictions, while ROC-AUC measures the overall discriminatory capability of the classifiers across different thresholds. The Precision-Recall curve also measures how well the algorithm can correctly classify spam and reduce the number of spam classifications.

II. Result Analysis And Discussion

This experimental testing of the effectiveness of the developed email spam classification models is done in a controlled computational environment. Python in Jupyter Notebook with Scikit-learn, LightGBM, Pandas, NumPy, and Matplotlib is used to implement the experiments. It is powered by a high-performance machine fitted with lots of cores, 16 GB of RAM and support for NVIDIA GPUs. The efficacy of the algorithm is evaluated using standard criteria for email spam detection classification.

A. Results of Proposed Models

The results of the suggested models for detecting spam in emails are shown in Table II. The MLP achieves an accuracy of 0.9839, while Extra Trees improves it to 0.9867 with high precision and specificity. LightGBM achieves an acc of 0.9846 with balanced recall and F1-score. Among all models, the VotingClassifier performs best, achieving the highest acc (0.9902), prec (0.9943), F1score (0.9900), and specificity (0.9945), along with a recall and sensitivity of 0.9858. These results demonstrate that combining the complementary predictions of the three base classifiers through soft voting provides more robust and reliable email spam classification than the individual models.

Table 2. *Performance of indivicual ML mpdels for email spam detection*

Model	ACC	PRE	REC	F1	SEN	SPE
MLP	0.9839	0.9843	0.9829	0.9836	0.9829	0.9850
Extra Trees	0.9867	0.9914	0.9815	0.9864	0.9815	0.9918
LightGBM	0.9846	0.9843	0.9843	0.9843	0.9843	0.9850
Voting Classifier	0.9902	0.9943	0.9858	0.9900	0.9858	0.9945

Fig 3 presents the confusion matrices of the four proposed models on the test dataset. MLP correctly classifies 720 non-spam and 690 spam emails, with 11 FP and 12 FN. Extra Trees correctly identifies 725 non-spam and 689 spam emails, producing only 6 FP and 13 FN. LightGBM correctly classifies 720 non-spam and 691 spam emails, with 11 FP and 11 FN. The Voting Classifier achieves the best overall classification, correctly identifying 727 non-spam and 692 spam emails while producing only 4 FP and 10 FN, demonstrating its superior ability to minimize misclassification.

Fig 4 and 5 present the ROC-AUC and Precision-Recall curves of the proposed models for spam detection. ROC analysis also demonstrates good discriminatory performance, with MLP and LightGBM having an AUC of 0.98, and Extra Trees and the Voting Classifier having an AUC of 0.99, which is a good separation between spam and non-spam emails. The Precision-Recall curves also reflect that all the models have high precision on most levels of recall, and that the Voting Classifier has a slightly more stable and high precision profile on the curve.

B. Comparison with Existing Models

Table 3 shows that the proposed models outperform the reported models in terms of the evaluated metrics. The proposed Voting Classifier also outperforms most of the existing methods with respect to acc (0.9902), prec (0.9943), rec (0.9858), and F1score (0.9900), and individual MLP, Extra Trees and LightGBM models also outperform most of the existing methods. The existing Random Forest, LR, NB, NN, VGG16, ResNet50, XGBoost and LSTM-GRU models perform comparatively less than the proposed model especially in terms of accuracy and F1 score. The results demonstrate that the suggested ensemble method outperforms the state-of-the-art in email spam detection.

Table 3. *Comparison between proposed and existing models for email spam detection*

Models/Measures		ACC	PRE	REC	F1
This work	MLP	0.9839	0.9843	0.9829	0.9836
	ET	0.9867	0.9914	0.9815	0.9864
	LightGBM	0.9846	0.9843	0.9843	0.9843
	Voting	0.9902	0.9943	0.9858	0.9900
Base model [15]	Random forest	0.9415	0.9371	0.9128	0.9247
M. Aleisa et al. [16]	LR	0.97	0.975	0.975	0.975
	NB	0.97	0.975	0.975	0.975
	NN	0.96	0.965	0.965	0.965
J. Jamil et al. [17]	VGG16	-	0.77	0.97	0.85
	ResNet50	-	0.93	0.97	0.95
	XGBoost	-	0.91	0.93	0.92
S. Saleem et al. [18]	LSTM-GRU	0.90	0. 83	0.84	0. 83

Advantage: The proposed work has several advantages for effective email spam detection system, such as high classification performance, less feature dimensionality, class balancing, and strong ensemble learning. The top 35 selected features contain the most informative features of the emails while also reducing the computational complexity, while SMOTE-ENN balances the classes and removes noisy samples. The soft voting of three classifiers (MLP, Extra Trees, LightGBM) takes advantage of their complementary learning abilities and decreases classification errors. The proposed Voting Classifier is also very specific (to reduce false positives) and performs better than some of the other models on several metrics.

C. Implications and Limitations

The proposed framework has significant practical potential to automatically and reliably classify emails as spam, as the use of feature selection, SMOTE-ENN, and Ensemble Learning can be used to effectively filter out unwanted emails while minimizing misclassifications. There are some limitations in the study, such as the use of the Spambase-UCI datasets, which may not be fully representative of the current email content, spam patterns, multilingual emails, and sophisticated adversarial attacks. Moreover, it is important to note selected features and model parameters need to be further validated on more large, varied and real-time email datasets.

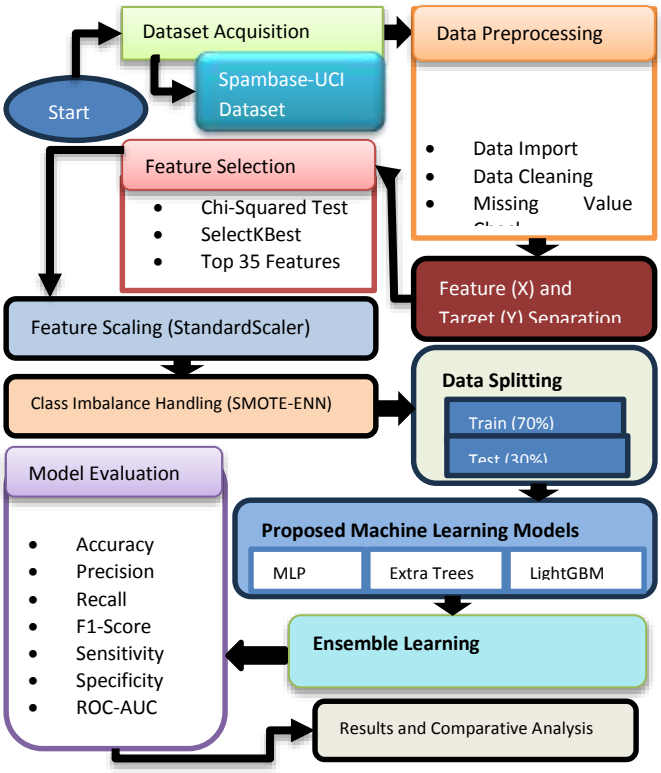


Figure 1. *Workflow of the Proposed Machine Learning-Based Email Spam Detection Framework*

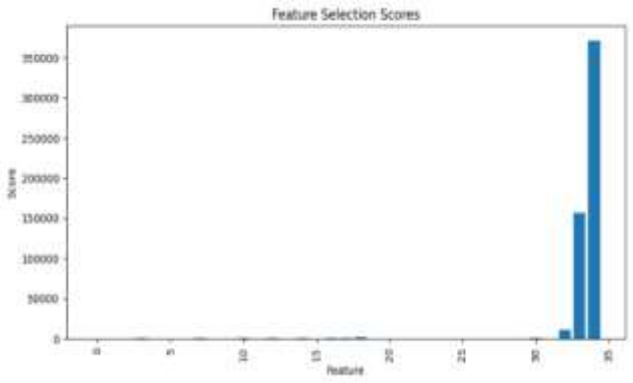


Figure 2. *Graph of feature selection score with Chi-Squared*

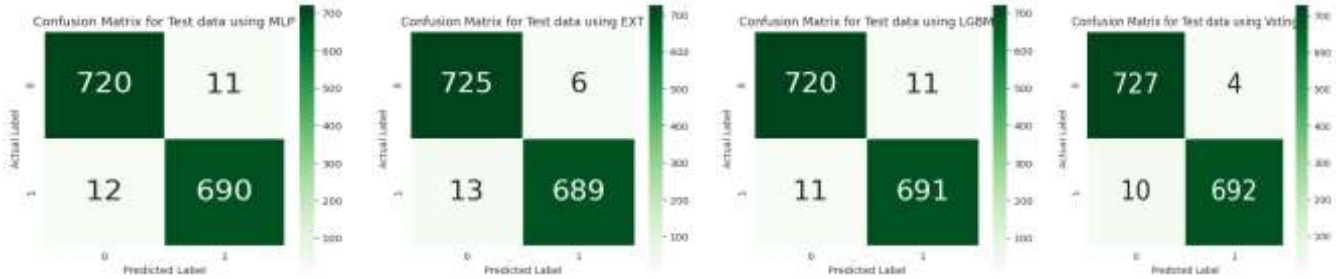


Figure 3. *Confusion Matrix of Proposed Models for Spam Detection*

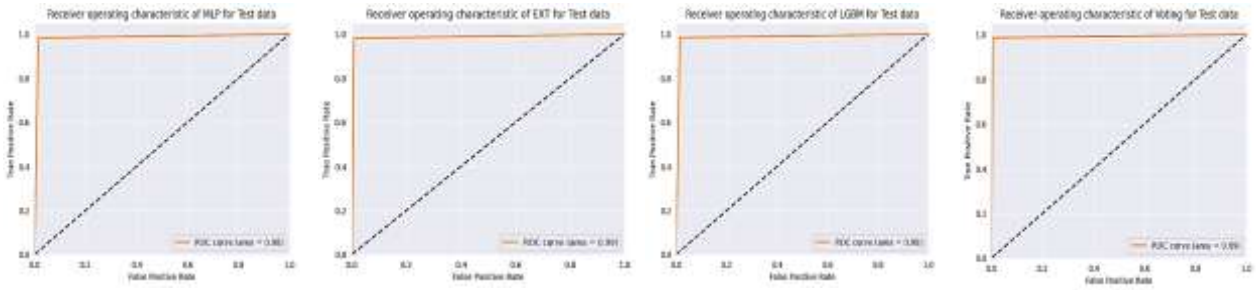


Figure 4. *ROC-AUC Curve of Proposed Models for Spam Detection*

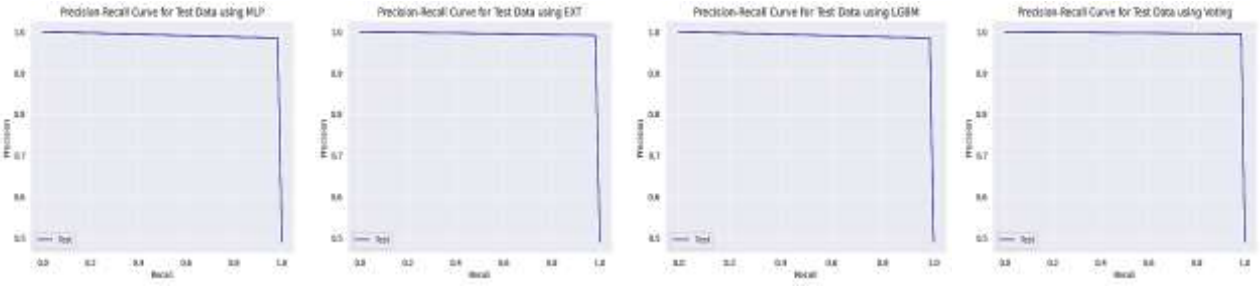


Figure 5. *Precision-Recall Curve of Proposed Models for Spam Detection*

4. Conclusions and Future works

Email spam detection is a key component of protecting consumers from unwanted and even harmful messages. The proposed study entails the development of an ML model for the successful detection of email spam using feature selection, data balancing and ensemble learning. The characteristics that were chosen were used to train the MLP, Extra Trees, and LightGBM models; the Soft Voting Classifier then merged the predictions of these three models. Outperforming the individual models and other current techniques, the testing findings show that the Voting Classifier attained the best overall performance with 99.02% accuracy. The outcomes indicate that the proposed framework is effective in accurately identifying spam emails. The framework will be tested in future on various datasets like SpamAssassin, Enron-Spam, Ling-Spam, TREC Spam, SMS Spam Collection. The advanced models such as XGBoost, CatBoost, CNN, LSTM, BiLSTM, BERT, and RoBERTa will be discussed. To enhance interpretability, XAI techniques like SHAP, LIME, and permutation feature importance will be introduced. The proposed framework can be further extended to address concept-drift adaptation, adversarial spam detection, multilingual email classification, and real-time deployment, which can further improve the practical applicability and robustness of the proposed framework.

Author Statements:

- **Ethical approval:** The conducted research is not related to either human or animal use.
- **Conflict of interest:** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper
- **Acknowledgement:** The authors declare that they have nobody or no-company to acknowledge.

- **Author contributions:** The authors declare that they have equal right on this paper.
- **Funding information:** The authors declare that there is no funding to be acknowledged.
- **Data availability statement:** The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.
- **Use of AI Tools:** The author(s) declare that no generative AI or AI-assisted technologies were used in the writing process of this manuscript.

References

[1] Y. Tian, X. Dai, Z. Li, H. Guo, and X. Mao, "Improving the accuracy of cybersecurity spam email detection using ensemble techniques: A stacking approach Machine learning for spam email detection," *PLoS One*, vol. 20, no. 9, p. e0331574, Sep. 2025, doi: 10.1371/journal.pone.0331574.

[2] M. Adnan, M. O. Imam, M. F. Javed, and I. Murtza, "Improving spam email classification accuracy using ensemble techniques: a stacking approach," *Int. J. Inf. Secur.*, vol. 23, no. 1, pp. 505–517, Feb. 2024, doi: 10.1007/s10207-023-00756-1.

[3] N. Ahmed, R. Amin, H. Aldabbas, D. Koundal, B. Alouffi, and T. Shah, "Machine Learning Techniques for Spam Detection in Email and IoT Platforms: Analysis and Research Challenges," *Secur. Commun. Networks*, vol. 2022, pp. 1–19, Feb. 2022, doi: 10.1155/2022/1862888.

[4] M. Crawford, T. M. Khoshgoftaar, J. D. Prusa, A. N. Richter, and H. Al Najada, "Survey of review spam detection using machine learning techniques," *J. Big Data*, vol. 2, no. 1, p. 23, Dec. 2015, doi: 10.1186/s40537-015-0029-9.

[5] C. P. Singh, "Enterprise-Grade AI-Driven Text Analytics and Insight Extraction Using Transformer-Based NLP Models," in *2026 14th International Symposium on Digital Forensics and Security (ISDFS)*, IEEE, Mar. 2026, pp. 1–6. doi: 10.1109/ISDFS69419.2026.11459059.

[6] K. Agarwal, P. Uniyal, S. Virendrasingh, S. Krishna, and V. Dutt, "Spam Mail Classification Using Ensemble and Non-Ensemble Machine Learning Algorithms," in *Lecture Notes in Networks and Systems*, 2021, pp. 179–189. doi: 10.1007/978-981-15-7106-0_18.

[7] A. M. John and S. S. Joseph, "A Lightweight and Interpretable Machine Learning Approach for SMS and Email Spam Detection," in *2026 International Conference on AI-Driven Solutions for Sustainable Smart Cities: Challenges and Opportunities (ADSSSC)*, IEEE, Apr. 2026, pp. 1–5. doi: 10.1109/ADSSSC67751.2026.11582785.

[8] T. Chanda, B. Mondal, S. Das, S. Banerjee, and B. Mondal, "An Explainable AI-Based Hybrid SVM–CNN Ensemble Model Implementation for Intelligent Email Spam Detection," in *2026 International Conference on Emerging Systems and Intelligent Computing (ESIC)*, IEEE, Feb. 2026, pp. 240–245. doi: 10.1109/ESIC68176.2026.11496294.

[9] V. Sowmitha, R. Deebika, K. Vinoth, S. Suryaprakash, K. Kalaivendhan, and V. Karthi, "Federated Learning in Email Server Authentication for Spam Mail Detection Using LSTM Algorithm," in *2025 International Conference on Smart and Sustainable Technology, INCSST 2025*, 2025. doi: 10.1109/INCSST64791.2025.11210294.

[10] L. Zou, "A Comparative Analysis of Count-Based and Inference-Based NLP Models in Spam Email Detection Task," in *2024 5th International Seminar on Artificial Intelligence, Networking and Information Technology, AINIT 2024*, 2024. doi: 10.1109/AINIT61980.2024.10581485.

[11] N. Ramojirao, S. Anusuya, Harshavardhini, and C. B. Sembuli, "Analysis of Email Spam Detection Using Naive Bayes and Support Vector Machine Classification Algorithms," in *ICETAS 2024 - 9th IEEE International Conference on Engineering Technologies and Applied Sciences*, 2024. doi: 10.1109/ICETAS62372.2024.11120193.

[12] A. Saini, K. Guleria, and S. Sharma, "Machine Learning Approaches for an Automatic Email Spam Detection," in *2023 International Conference on Artificial Intelligence and Applications, ICAIA 2023 and Alliance Technology Conference, ATCON-1 2023 - Proceeding*, 2023. doi: 10.1109/ICAIA57370.2023.10169201.

[13] V. I. Del Rosario, B. D. P. Fernandez, and D. A. Padilla, "Email Spam Classification using DistilBERT," in *2023 IEEE 15th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management, HNICEM 2023*, 2023. doi: 10.1109/HNICEM60674.2023.10589211.

[14] J. Hopkins, M., Reeber, E., Forman, G., & Suermondt, "UCI Machine Learning Repository," *Kaggel*, 1999, [Online]. Available: https://archive.ics.uci.edu/dataset/94/spambase

[15] A. Saini, K. Guleria, and S. Sharma, "Machine Learning Approaches for an Automatic Email Spam Detection," in *2023 International Conference on Artificial Intelligence and Applications (ICAIA) Alliance Technology Conference (ATCON-1)*, IEEE, Apr. 2023, pp. 1–5. doi: 10.1109/ICAIA57370.2023.10169201.

[16] M. H. Alsuwit, M. A. Haq, and M. A. Aleisa, "Advancing Email Spam Classification using Machine Learning and Deep Learning Techniques," *Eng. Technol. Appl. Sci. Res.*, vol. 14, no. 4, pp. 14994–15001, Aug. 2024, doi: 10.48084/etasr.7631.

[17] M. Jamil, H. Mihajloska Trpcheska, A. Popovska-Mitrovikj, V. Dimitrova, and R. Creutzburg, "Advancing Image Spam Detection: Evaluating Machine Learning Models Through Comparative

Analysis," *Appl. Sci.*, vol. 15, no. 11, p. 6158, May 2025, doi: 10.3390/app15116158.

[18] S. Saleem, Z. U. Islam, S. S. U. Hasan, H. Akbar, M. F. Khan, and S. A. Ibrar, "Spam Email Detection Using Long Short-Term Memory and Gated Recurrent Unit," *Appl. Sci.*, vol. 15, no. 13, p. 7407, Jul. 2025, doi: 10.3390/app15137407