International Journal of Engineering Research and Science & Technology
ISSN 2319-5991 www.ijerst.com

Peer Reviewed, Referred & Indexed Journal

Research Paper

Ensemble Learning-Based Approach for Email Spam Detection

Mohammed Hussain Afzaal,
PG Scholar, Department of Computer Science and Engineering,
Shadan College of Engineering and Technology, Hyderabad, Telangana, India - 500086
Email: mohammedhussainafzaal@gmail.com

Subramanian K.M
'Professor, Department of Computer Science and Engineering,
Shadan College of Engineering and Technology, Hyderabad, Telangana, India - 500086
Email: kmsubbu.phd@gmail.com

Md. Ateeq Ur Rahman
Professor, Department of Computer Science and Engineering:
Shadan College of Engineering and Technology, Hyderabad, Telangana, India - 500086
Email: mail_to_ateeq@yahoo.com

***Abstract*— Email spam continues to pose significant security threats, productivity losses, and user inconveniences, making accurate and efficient detection increasingly essential. Traditional machine learning techniques, though widely applied, often struggle with limited accuracy and generalization, necessitating more robust detection strategies. The Enron Email Dataset, sourced from Kaggle, provides a comprehensive collection of email messages, encompassing both spam and legitimate emails. The dataset includes thousands of emails with varied content, formats, and lengths, offering a realistic benchmark for testing spam detection methodologies. The approach employs extensive preprocessing steps, including encoding, data cleaning, tokenization, stopword removal, stemming or lemmatization, TF-IDF vectorization, and oversampling with SMOTE to balance class distribution. Multiple algorithms are applied, including Decision Tree, Random Forest, Naive Bayes, Extra Trees, MLP, SVM, Proposed_Voting, and Proposed_Stacking, with Proposed_Stacking demonstrating superior predictive capability. To further enhance detection performance, Extension_Voting and Extension_Stacking models integrate ensemble strategies to optimize accuracy, precision, recall, and F1-score, achieving the highest result of 98.8%. Along with that, XAI SHAP is utilized as another enhancement for interpretability, and the Flask Framework is employed for the user interface, enabling users to provide inputs and obtain predictions using the trained models. These enhancements suggest promising avenues for more reliable, scalable, and real-time spam detection solutions.**

Keywords*— *Email Spam Detection, Machine Learning, Ensemble Learning, Support Vector Classifier, Multilayer Perceptron, Extra Tree Classifier.

I. INTRODUCTION

The exponential advancement of digital technologies has profoundly transformed communication and information exchange across the globe. Among the various innovations that underpin this transformation, electronic mail (email) has emerged as one of the most prevalent and indispensable forms of digital communication. Owing to its rapid transmission, cost efficiency, and global accessibility, email has become the cornerstone of both personal and professional correspondence in the digital era [1]. Recent statistical reports indicate that the number of global email users has experienced remarkable growth, reaching approximately 4.4 billion in 2024 and projected to surpass 4.8 billion by 2027 [2]. This extensive adoption reflects the pivotal role of email in the daily operations of individuals, enterprises, and institutions, thereby making it a critical infrastructure of the information age. However, the increasing dependence on email communication has also led to the emergence of substantial cybersecurity challenges, notably the proliferation of spam and malicious email activities [3].

Despite continuous advancements in email filtering technologies, spam remains one of the most persistent and evolving threats within cyberspace. Cybercriminals exploit spam as an attack vector to distribute phishing content, spread malware, and execute large-scale social engineering campaigns [4]. It is estimated that nearly 90% of all cyberattacks originate from malicious or deceptive emails, underscoring the scale and gravity of the problem [5]. The enormous volume of spam traffic not only imposes financial burdens on organizations but also consumes significant network bandwidth and computational resources. Traditional email filtering techniques, primarily reliant on rule-based or statistical methods, often struggle to adapt to the ever-changing characteristics of spam messages [6]. Furthermore, conventional detection frameworks tend to exhibit reduced accuracy and scalability when exposed to diverse datasets, dynamic language patterns, and sophisticated obfuscation strategies [7]. This highlights a clear research gap, wherein existing solutions fail to provide a robust and adaptive mechanism capable of addressing the continuously evolving nature of spam threats in real-world contexts.

In light of these challenges, this research aims to propose an intelligent and adaptive framework for enhancing the accuracy, efficiency, and reliability of email spam detection. The study explores advanced data-driven methodologies that can learn from dynamic email environments and identify latent patterns that distinguish legitimate messages from

spam. Unlike conventional approaches that often rely on static classification mechanisms, the proposed framework emphasizes the integration of intelligent learning paradigms capable of improving detection performance across diverse and evolving datasets [8]. By adopting a structured and scalable design, the framework seeks to optimize classification efficiency while maintaining interpretability and generalizability—two crucial aspects often overlooked in prior research efforts [9].

The significance of this study lies in its potential to contribute meaningfully to both academic research and real-world cybersecurity practices. By improving the reliability of automated spam detection systems, the proposed approach can enhance email security infrastructure, reduce operational costs associated with spam filtering, and safeguard users from phishing, identity theft, and data breaches. Moreover, the framework can serve as a foundation for developing more resilient and adaptive cybersecurity models capable of countering emerging threats in the digital communication landscape [10]. Overall, this research aspires to bridge the gap between theoretical advancements and practical implementations in spam detection, ultimately strengthening the global effort toward secure and trustworthy digital communication.

II. Related Work

Email spam detection has attracted sustained attention in cybersecurity and machine learning research due to the rapid growth and increasing sophistication of unsolicited and malicious emails. Early and contemporary studies have focused on improving detection accuracy, reducing false positives, and enhancing computational efficiency. Gupta and Agarwal [11] proposed a hybrid machine learning framework that combined multiple classifiers and linguistic features, achieving improved accuracy and reduced false positives, though scalability to large and evolving datasets remained limited. Patel and Desai [12] employed convolutional neural networks to learn hierarchical text representations, demonstrating superior performance over traditional classifiers; however, their approach required substantial computational resources and large labeled datasets.

Hybrid and ensemble-based techniques have also been widely explored. Singh and Kumar [13] integrated ensemble learning with feature selection to improve spam classification precision, highlighting the importance of relevant feature optimization, but their model showed reduced generalization on heterogeneous datasets. Zhang and Wang [14] applied recurrent neural networks to capture sequential and contextual dependencies in email text, revealing strong performance for complex spam patterns while encountering difficulties with long and highly variable messages. Tusher et al. [15] provided a comprehensive survey of spam detection techniques, emphasizing that despite advancements in machine and deep learning, many models remain vulnerable to concept drift, emerging spam strategies, and limited interpretability.

Several studies focused on enhancing traditional machine learning algorithms. Nagaraj et al. [16] demonstrated the efficiency of Naïve Bayes classifiers for spam detection but noted sensitivity to class imbalance and limited feature diversity. Mythili et al. [17] introduced a temporal Naïve Bayes model that incorporated time-based patterns, achieving modest improvements but struggling with non-stationary spam behavior. Doshi et al. [18] proposed a dual-layer architecture combining content-based and behavioral features for spam and phishing detection, which improved robustness at the cost of increased computational complexity.

Recent research emphasizes comparative evaluation and optimization of machine learning models. Sonare et al. [19] analyzed multiple classifiers, concluding that preprocessing quality and feature engineering significantly influence performance. Ch et al. [20] compared supervised learning approaches such as decision trees and support vector machines, reporting promising results but inconsistent accuracy across datasets with diverse linguistic structures and spam distributions.

Despite substantial progress, existing studies continue to face challenges related to adaptability, scalability, and interpretability in dynamic spam environments. Many models are sensitive to dataset imbalance, linguistic diversity, and evolving attack strategies, while few effectively balance predictive performance with explainability. Addressing these limitations, the present study proposes an adaptive framework that integrates multiple learning paradigms to improve accuracy, robustness, and transparency, thereby enhancing the effectiveness of email spam detection in real-world scenarios.

III. Materials And Methods

The proposed email spam detection system is developed to accurately classify spam and legitimate emails using the Enron Email Dataset, which contains diverse and realistic email content. The system follows a standardized machine learning workflow in which email text is transformed into discriminative numerical representations using TF–IDF feature extraction after text normalization. To handle class imbalance and improve generalization, SMOTE-based oversampling is applied during training. Multiple supervised classifiers, including Decision Tree [29], Naive Bayes [28], Random Forest [22], Extra Trees [25], MLP [24], and SVM [23], are first evaluated individually and then integrated into ensemble frameworks. A Proposed Voting Classifier and a Proposed Stacking Classifier combine these conventional learners to enhance prediction stability and reduce model bias. In addition, advanced Voting and Stacking Classifiers employing AdaBoost, XGBoost, and CatBoost are incorporated to capture complex nonlinear patterns and further improve detection performance [26]. Model interpretability is ensured through XAI techniques using LIME and SHAP, while deployment using the Flask framework supports real-time, scalable, and explainable spam detection [27].

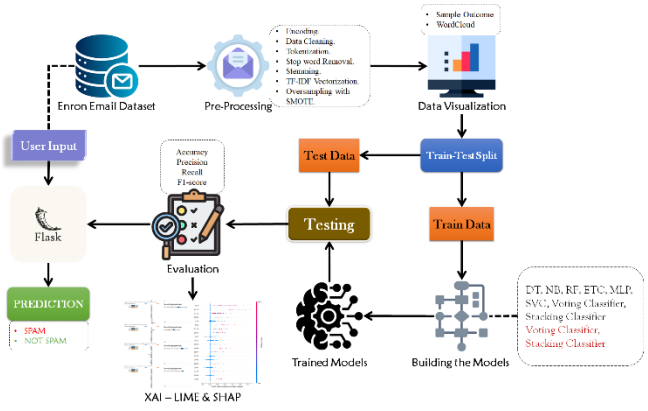

Fig. 1. System Architecture

The system architecture for email spam detection comprises multiple interconnected layers as shown in fig.1, beginning with data ingestion from the Enron Email Dataset followed by preprocessing and feature extraction using TF-IDF vectorization. Balanced data is achieved through SMOTE to mitigate class imbalance. Machine learning and ensemble models, including both proposed and advanced Voting and Stacking classifiers, are employed for accurate classification. XAI-SHAP enhances interpretability by explaining feature influences, while the Flask-based interface enables real-time user interaction, ensuring scalable, transparent, and efficient spam detection performance [30].

a) Dataset Collection

The Enron Email Dataset is employed in this study to develop and evaluate the proposed email spam detection system. The dataset consists of 5,572 email messages with two primary attributes: message content and class label. Emails are categorized into spam and non-spam (ham), exhibiting a natural class imbalance reflective of real-world scenarios. The dataset contains diverse textual patterns, lengths, and linguistic structures, making it suitable for robust spam classification. Its realistic distribution and publicly validated usage support reliable performance evaluation and generalization of machine learning models.

	Category	Message
0	ham	Go until jurong point, crazy.. Available only ...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives aro...

Fig.2 Enron Email Dataset

b) Pre-Processing

The preprocessing phase transforms raw email data into structured, analyzable form through systematic cleaning, normalization, feature extraction, and sampling, ensuring improved data quality, balance, and reliability for accurate spam classification.

Data Pre-processing: The initial data preprocessing phase involves transforming the raw email dataset into a structured form suitable for analysis. The textual data and categorical labels are refined by encoding class variables into numerical format, distinguishing spam and non-spam messages. This transformation ensures compatibility with machine learning algorithms that require numeric inputs. Proper preprocessing facilitates efficient data handling, minimizes inconsistencies, and establishes a foundational structure for subsequent analytical and classification procedures within the spam detection framework.

Data Cleaning: The data cleaning stage focuses on improving the quality and uniformity of textual data. Unwanted components such as HTML tags, URLs, digits, and special symbols are eliminated, and all characters are converted to lowercase to maintain uniformity. These cleaning measures reduce noise and remove irrelevant information, allowing models to focus on meaningful textual patterns. This step enhances the reliability of feature extraction and contributes to higher accuracy and consistency in spam classification performance.

Tokenization: Tokenization converts cleaned textual data into smaller linguistic units known as tokens, typically representing individual words. This process enables the segmentation of continuous text into analyzable components that can be quantitatively assessed. Tokenization forms the foundation for various natural language processing tasks, as it allows the system to capture word-level features and frequency distributions. Consequently, it enables a more structured representation of data, improving the precision and interpretability of subsequent text-based analyses.

Stopword Removal: Stopword removal eliminates commonly occurring words such as “the,” “is,” and “and” that contribute little to semantic meaning. By removing these non-informative elements, the model can focus on contextually relevant terms that distinguish spam from non-spam messages. This reduction in data dimensionality improves processing efficiency and strengthens the discriminative capability of the model. Consequently, stopword removal enhances both computational performance and classification accuracy within the email spam detection system.

Text Normalization (Stemming and Lemmatization): Text normalization involves refining tokens through stemming and lemmatization to reduce words to their base or root forms. This harmonizes morphological variations, ensuring that words with similar meanings are treated equivalently during analysis. Normalization helps reduce redundancy, enhances vocabulary consistency, and minimizes feature sparsity within the dataset. This linguistic simplification strengthens feature representation and ensures that the classification models can identify conceptual similarities between words, thereby improving overall system robustness and interpretability.

Feature Engineering using TF-IDF: Feature engineering is accomplished through Term Frequency–Inverse Document Frequency (TF-IDF) vectorization, which transforms textual information into numerical feature representations. This approach quantifies the significance of each term relative to its occurrence across documents, highlighting contextually important words. By capturing both local and global term importance, TF-IDF provides a balanced representation of text data. This transformation enables the machine learning models to detect subtle distinctions between spam and legitimate messages, improving classification effectiveness.

Sampling using SMOTE: The Synthetic Minority Oversampling Technique (SMOTE) is employed to address class imbalance between spam and non-spam categories. By synthetically generating new instances of the minority class, SMOTE ensures a balanced dataset that prevents model bias toward majority samples. This balance enhances the generalization capability and fairness of the classification models. Consequently, SMOTE plays a vital role in improving model performance, accuracy, and stability in detecting less frequent but critical spam instances.

c) Training and Testing:

The dataset is divided into training and testing subsets to evaluate model generalization and predictive capability. The training set is used to learn patterns and relationships, while the testing set assesses performance on unseen data. This separation prevents overfitting and ensures that the model's accuracy reflects its real-world applicability. Employing an appropriate train-test ratio supports reliable validation, enabling consistent and unbiased performance assessment of the spam detection algorithms.

d) Algorithms

Decision Tree: Decision Tree is a supervised classification method that learns hierarchical decision rules through recursive data partitioning. It efficiently captures nonlinear patterns, offers transparent decision-making, and provides fast, reliable predictions, making it a strong baseline learner for ensemble-based classification tasks.

$$I\,(i) = 1 - \sum_{i=1}^{k} p_i^2 \quad (1)$$

Naïve Bayes: Naive Bayes is a probabilistic classifier that estimates class membership using conditional probabilities under feature independence assumptions. It is computationally efficient, well-suited for high-dimensional feature spaces, and contributes to stable performance and strong generalization in text-based classification problems.

$$P(A|B) = \frac{P(B|A).P(A)}{P(B)} \quad (2)$$

Random Forest: Random Forest is an ensemble learning technique that aggregates predictions from multiple decision trees trained on randomized data subsets and features. This diversity reduces overfitting, enhances robustness, and improves classification accuracy and generalization across complex and large-scale datasets.

$$Gini = 1 - \sum_{i=1}^{C} (Pi)^2 \quad (3)$$

Extra Trees Classifier: The Extra Trees Classifier builds an ensemble of decision trees using highly randomized feature splits. Increased randomness improves model diversity, reduces variance, and enhances robustness, resulting in accurate and computationally efficient classification performance for complex feature distributions.

Multilayer Perceptron: Multilayer Perceptron is a feedforward neural network capable of modeling complex nonlinear relationships through layered transformations. Its ability to learn intricate patterns enhances predictive accuracy, adaptability, and generalization, particularly in high-dimensional classification scenarios.

$$\hat{y} = f\left(W^L f\left(W^{L-1} \cdots f(W^1 X + b^1) + b^{(L-1)}\right) + b^L\right) \quad (4)$$

Support Vector Classifier: Support Vector Classifier constructs an optimal decision boundary by maximizing the margin between classes. Through kernel-based transformations, it effectively handles nonlinear feature spaces, providing robust classification performance and strong resistance to overfitting in high-dimensional data.

Voting Classifier: The Voting Classifier is an ensemble strategy that aggregates predictions from multiple base learners through majority or weighted voting. By combining diverse decision patterns, it reduces individual model bias and improves overall robustness, stability, and classification accuracy.

$$\hat{y} = argmax_c \left(\sum_{i=1}^{n} II\,(\hat{y}_i = c) \right) \quad (5)$$

Stacking Classifier: The Stacking Classifier integrates multiple base models by training a meta-learner on their predictions. This layered approach captures complementary strengths among classifiers, enhances predictive performance, and improves generalization in complex classification environments.

$$\hat{y} = g(Y_{base}) = g\left(f_1(x), f_2(x), \ldots, f_m(x)\right) \quad (6)$$

Extension Voting Classifier: The Extension Voting Classifier enhances traditional voting by assigning optimized weights to base learners based on performance. This strategy prioritizes stronger models, reduces prediction bias, and delivers more consistent and reliable classification outcomes across diverse data distributions.

Extension Stacking Classifier: The Extension Stacking Classifier improves conventional stacking by strengthening model diversity and optimizing meta-level learning. By effectively combining multiple classifier outputs, it enhances robustness, reduces overfitting, and achieves superior generalization in challenging classification tasks.

e) Integration of XAI and Flask Framework:

To enhance interpretability and transparency, Explainable Artificial Intelligence (XAI) tools such as LIME and SHAP are integrated into the spam detection framework. LIME provides local explanations by identifying the most influential words contributing to each individual prediction, allowing users to understand model reasoning on a case-by-case basis. SHAP offers a global perspective by quantifying the overall impact of features across predictions, visualizing how each term influences spam or non-spam classification outcomes.

The trained ensemble models and XAI components are deployed using the Flask web framework, enabling real-time prediction and explanation through a user-friendly interface. This integration facilitates practical accessibility, allowing users to input email text, receive classification results instantly, and visualize interpretability outputs interactively, ensuring both operational efficiency and model transparency.

IV. Experimental Results

Accuracy: The accuracy of a test is its ability to differentiate the patient and healthy cases correctly. To estimate the accuracy of a test, we should calculate the proportion of true positive and true negative in all evaluated cases. Mathematically, this can be stated as:

$$Accuracy = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{TN} + \mathrm{FN}} \quad (7)$$

Precision: Precision evaluates the fraction of correctly classified instances or samples among the ones classified as positives. Thus, the formula to calculate the precision is given by:

$$Precision = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} (8)$$

Recall: Recall is a metric in machine learning that measures the ability of a model to identify all relevant instances of a particular class. It is the ratio of correctly predicted positive observations to the total actual positives, providing insights into a model's completeness in capturing instances of a given class.

$$Recall = \frac{\text{TP}}{\text{TP} + \text{FN}} (9)$$

F1-Score: F1 score is a machine learning evaluation metric that measures a model's accuracy. It combines the precision and recall scores of a model. The accuracy metric computes how many times a model made a correct prediction across the entire dataset.

$$F1\ Score = 2 * \frac{Recall\ \text{X}\ Precision}{Recall + Precision} * 100 (10)$$

Table. 1. Performance Evaluation

ML Model	Accuracy	Precision	Recall	F1-Score
Decision Tree	0.921	0.921	0.921	0.921
Random Forest	0.926	0.926	0.926	0.935
Naïve Bayes	0.972	0.972	0.972	0.972
Extra Trees	0.932	0.931	0.932	0.940
MLP	0.978	0.978	0.978	0.978
SVM	0.978	0.978	0.978	0.978
Proposed Voting	0.953	0.953	0.953	0.957
Proposed Stacking	0.985	0.985	0.985	0.986
Extension Voting	0.988	0.988	0.988	0.988
Extension Stacking	0.987	0.987	0.987	0.987

The Table 1 presents model performance comparisons using Accuracy, Precision, Recall, and F1-Score metrics, where the Extension_Voting model outperforms all others, achieving the highest overall evaluation scores.

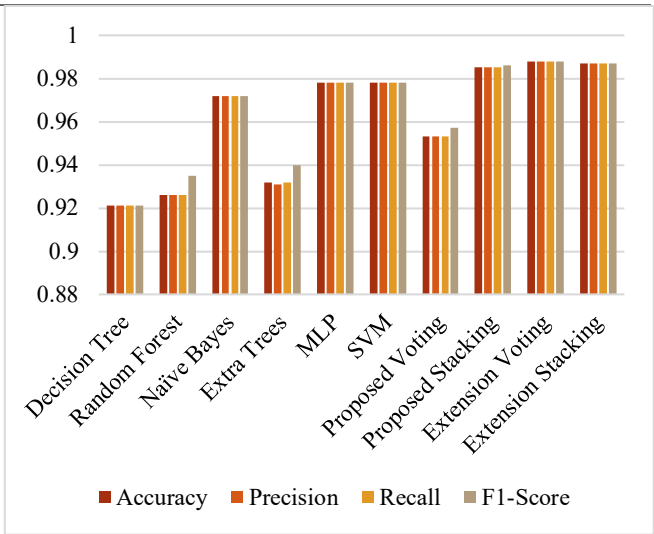

Fig. 3. Comparison of ML Models

The Fig.3 visually illustrates Accuracy, Precision, Recall, and F1-Score across models, highlighting that Extension_Voting consistently achieves superior performance, outperforming all other classifiers in all evaluation metrics.

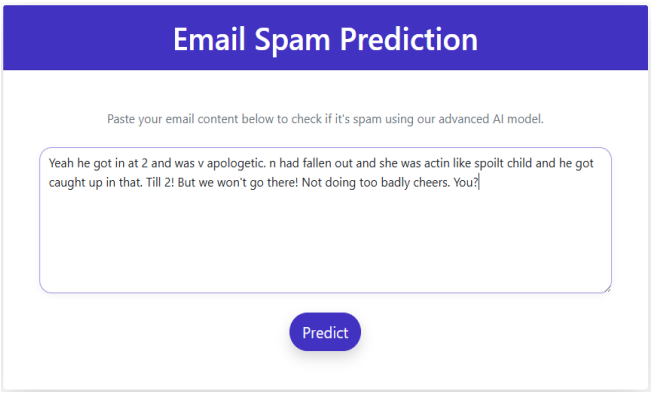

Fig. 4 Enter Email Content – Case 1

In Fig.4, the user enters email content into the text box, allowing the AI model to analyze and predict potential spam.

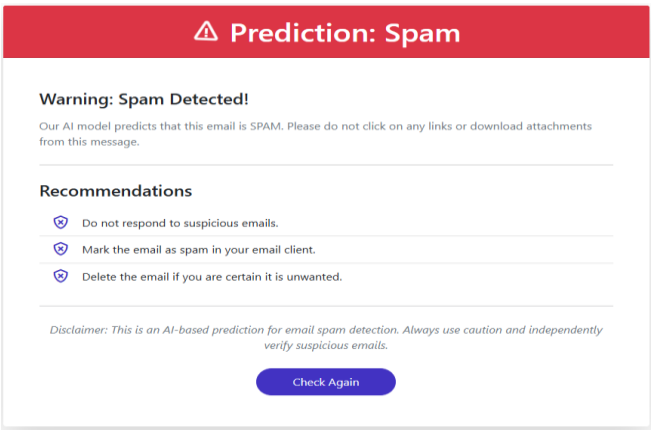

Fig. 5 Predicated Result - Spam

In Fig.5, the system analyzes the submitted email text and displays a Spam prediction along with recommended safety actions for users.

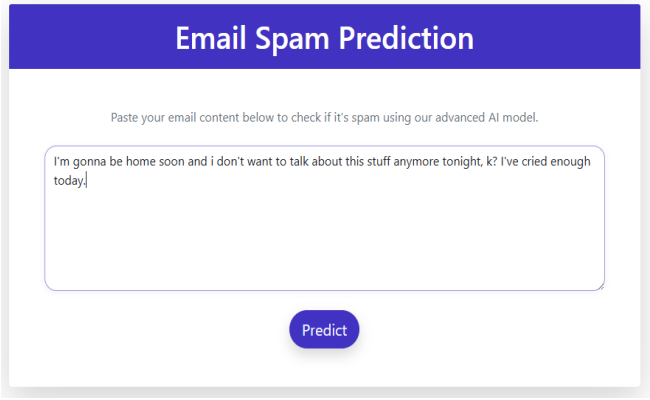

Fig.6 Enter Email Content – Case 2

In Fig.6, the user enters email content into the text box, allowing the AI model to analyze and predict potential spam.

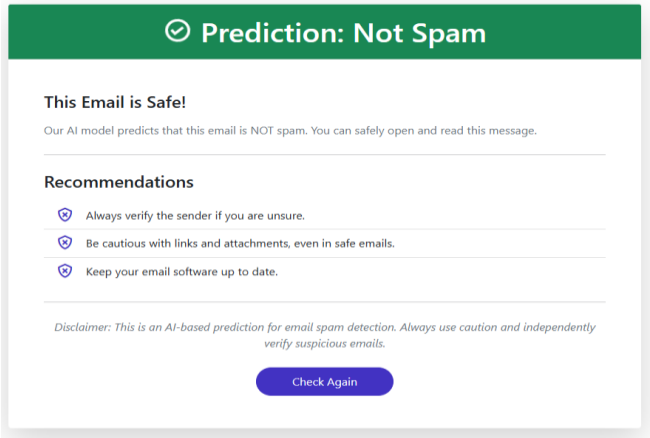

Fig.7 Predicated Result – Not Spam

In Fig.7, the system analyzes the submitted email text and displays a Not Spam prediction along with recommended safety actions for users.

V. Conclusion

In conclusion, the email spam detection system effectively leverages the Enron Email Dataset, which provides a comprehensive and diverse collection of spam and legitimate emails, to develop a robust and scalable solution. The methodology incorporates extensive preprocessing techniques, including encoding, data cleaning, tokenization, stopword removal, stemming or lemmatization, and TF-IDF vectorization, ensuring high-quality feature extraction. Oversampling with SMOTE addresses class imbalance, enabling balanced training across spam and non-spam classes. Multiple machine learning algorithms, including Decision Tree, Random Forest, Naive Bayes, Extra Trees, MLP, SVM, Proposed_Voting, and Proposed_Stacking, are employed to optimize detection performance, with ensemble-based approaches improving robustness and generalization. To further enhance accuracy and interpretability, the system integrates Voting Classifier and Stacking Classifier using AdaBoost, XGBoost, and CatBoost, along with XAI LIME & SHAP for feature-level explainability. The Flask Framework provides a user-friendly interface for real-time input and prediction. The integrated approach achieves a high accuracy, precision, recall, and F1-score of 98.8%, demonstrating superior performance, reliability, and practical applicability for email spam detection across diverse datasets and real-world scenarios.

VI. Future Scope

The future scope of this system involves integrating deep learning architectures such as LSTM and BERT for improved semantic understanding of email content. Incorporating real-time data streams can enhance adaptability against evolving spam patterns. Further, expanding the dataset with multilingual and domain-specific emails can improve generalization across diverse communication platforms. Enhanced visualization techniques with advanced XAI tools can offer deeper interpretability for end-users. Additionally, deploying the model as a scalable web service through cloud-based APIs can ensure wider accessibility, faster processing, and seamless integration with enterprise email systems for proactive spam mitigation.

References

[1] Al-augby, S., Alyasiri, H., Abdulkadhim, F. G., & Oleiwi, Z. C. (2025). A stacked ensemble classifier for email spam detection via an evolutionary algorithm. Mesopotamian Journal of CyberSecurity, 5(2), 657–670.

[2] Wang, L. (2025). Spam email detection using Naïve Bayes classifier. In ITM Web of Conferences (Vol. 70, p. 04028). EDP Sciences.

[3] El Bitar, I., Abbas, N., Raad, M., Shmouri, F., & El Khechen, F. (2025, February). Ensemble learning-based approach for email spam detection. In 2025 5th IEEE Middle East and North Africa Communications Conference (MENACOMM) (pp. 1-6). IEEE.

[4] Ahmed, W., & Hammad, M. (2026). Hybrid machine learning approach for phishing eMail detection using NLP and ensemble models. In Cybersecurity, cybercrimes, and smart emerging technologies (pp. 72-81). CRC Press.

[5] Al-augby, S., Alyasiri, H., Abdulkadhim, F. G., & Oleiwi, Z. C. (2025). A Stacked Ensemble Classifier for Email Spam Detection via an Evolutionary Algorithm. Mesopotamian Journal of CyberSecurity, 5(2), 657-670.

[6] Yeasmin, M. N., Refat, M. A. R., Singh, B. C., Alom, Z., Aung, Z., & Azim, M. (2026). EnLeM: ensemble learning-based model to detect phishing websites. EURASIP Journal on Information Security, 2026(1), 1.

[7] Alam, S., Jameel, A., Parveen, Z., & Alnfrawy, E. (2025). SHRED: An Ensemble-Based Machine Learning Model to Sift Email Messages for Real-Time Spam Detection. IEEE Access.

[8] Arfaoui, N. (2026). Enhancing Machine Learning Algorithms for Imbalanced Data. A Case Study: Spam Detection. IEEE Access.

[9] Jain, V. (2025, April). Intelligent email spam detection: A machine learning-based approach. In 2025 5th International Conference on Trends in Material Science and Inventive Materials (ICTMIM) (pp. 1574-1579). IEEE.

[10] Korkmaz, Y. (2026). Fraud E-mail detection using intelligent algorithms: Comparison of traditional approaches with deep learning techniques. Information Processing & Management, 63(2), 104416.

[11] Umamaheswari, T. S., & Umaselvi, M. (2025). ENHANCED ENSEMBLE CLASSIFICATION TECHNIQUES FOR ACCURATE SPAM DETECTION IN E-MAIL COMMUNICATIONS. ICTACT Journal on Soft Computing, 16(1).

[12] Das, L., Ahuja, L., & Pandey, A. (2026). Machine Learning-Based Optimization Algorithms for Spam SMS Classification. Engineering, Technology & Applied Science Research, 16(1), 32608-32618.

[13] Yadav, D. K., Raj, A., Rajlakshmi, Kumar, N., & Kumari, R. (2025). Enhancing Email Security: A Real‐Time Machine Learning‐Based Spam Detection System. Internet Technology Letters, 8(5), e618.

[14] Soysaldı Şahin, M., Şahin, D. Ö., & Salah, A. F. (2026). Revisiting SMS Spam Detection: The Impact of Feature Representation on Classical Machine Learning Models. Electronics, 15(4), 894.

[15] Hadi, M. T., & Baawi, S. S. (2025, March). Stacking ensemble learning with recursive feature elimination technique for email spam detection. In AIP Conference Proceedings (Vol. 3264, No. 1, p. 030029). AIP Publishing LLC.

[16] Abualhaj, M. M., Al-Khatib, S. N., Hiari, M. O., Shambour, Q. Y., Al-Allawee, A., Almomani, O., ... & Anbar, M. (2026). An efficient feature selection technique to enhance spam email detection. TEM Journal, 15(1), 91.

[17] Abid, A., Arsalan, M., Ahmed, E., Ahmed, M., & Shahid, M. M. A. (2025, November). An Ensemble and Deep Learning Based Approach for SMS Spam Classification. In 2025 International Conference on Engineering & Computing Technologies (ICECT) (pp. 1-8). IEEE.

[18] James, A. (2026). ADVANCED MACHINE LEARNING FRAMEWORKS FOR SCALABLE VOIP SPAM DETECTION IN CLOUD ENVIRONMENTS.

[19] Akeel, M., Butt, K. K., Javed, K., Tariq, M., & Yousaf, M. (2025). Email Spam Detection Using Machine Learning with Optimized Feature Engineering and Classification Techniques. Journal of Computing & Biomedical Informatics, 10(01).

[20] Klaus, H., & Ade, E. (2026). Machine learning-based phishing detection systems.

[21] Fareed, B., Din, G. M., Khan, M. R. A., Fatima, T., & Shahid, S. (2025). Machine Learning-Based Email Spam Detection: Accuracy, Overfitting and Robustness Analysis. European Journal of Applied Science, Engineering and Technology, 3(6), 99-108.

[22] Corpuz, J. E. A., Diaz, S. Q., Palafox, L. B. M., Tan, M. C. T., & Solomon, K. Y. C. (2026, April). Machine Learning-Based Detection of SMS Phishing in the Philippine Context. In Workshop on Computation: Theory and Practice (WCTP 2025) (pp. 553-562). Atlantis Press.

[23] Gond, S. P., Yadav, S. S., Alam, M. A., & Sahoo, S. (2025, March). Email Spam Detection Ensemble Methods. In 2025 3rd International Conference on Disruptive Technologies (ICDT) (pp. 1561-1567). IEEE.

[24] Horapeti, B. S., Rahaman, M., Gaurav, A., Arya, V., Chui, K. T., & Gupta, B. B. (2026). Hybrid Machine Learning-Based Detection of Phishing Attacks through E-mails and SMSs in Electronic Vehicles. SN Computer Science, 7(1), 94.

[25] Suman, S. K., Sahu, P., Khatkar, M., Haribabu, K., Poongodi, S., & Durani, H. (2025, August). Stochastic Gradient Boosting for Robust Twitter Spam Detection: An Ensemble Learning Approach. In 2025 World Skills Conference on Universal Data Analytics and Sciences (WorldSUAS) (pp. 1-6). IEEE.

[26] Chakravorty, A., Price, M., Elsayed, N., & ElSayed, Z. (2026). Context-Aware Phishing Email Detection Using Machine Learning and NLP. arXiv preprint arXiv:2603.27326.

[27] Devi, N. M., Asha, V., Dhiman, D., Patil, P., Utkarsh, P., & Poojari, V. (2025, June). Email security enhancement via ML-based spam filter techniques. In 2025 6th International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV) (pp. 336-340). IEEE.

[28] Sharma[1], D., Kaur, S., Bansal[1], D., & Dabra, M. (2026, June). A Comparative Study of Machine Learning Approaches. In Proceedings of the Conference on Bridging Engineering Disciplines with AI and Machine Learning (BEDAIML 2026) (p. 12). Springer Nature.

[29] Das, S. S., Samantaray, A., Palai, C., & Jena, P. K. (2025, December). Spam Email Detection with TF-IDF Features Using Ensemble Learning. In 2025 OITS International Conference on Information Technology (OCIT) (pp. 13-17). IEEE.

[30] Basheer, S., Hayat, I., Leghari, A. A., Phul, Z., & Quadri, S. K. A. (2026). CYBERSECURITY ANALYSIS OF PHISHING ATTACKS USING MACHINE LEARNING-BASED DETECTION AND PREVENTION TECHNIQUES. Spectrum of Engineering Sciences, 4(3), 2180-2193.