International Journal of Information Technology (IJIT) – Volume 11 Issue 2, Mar – Apr 2025

RESEARCH ARTICLE OPEN ACCESS

Email Inbox Management Using Machine Learning and Artificial Intelligence

Siva Rama Krishna Konolla[1], Kameswari Gorrela[2], Uday Sagar Reddy Maddi[3], Amaranadha Reddy ThallapuReddy[4], Srinivas Reddy Nadikattu[5], Gnana Manikanta Anumukonda[6]

[1,2,3,4,5,6]Department of Computer Science and Engineering, Lingaya's Institute of Management and Technology, Vijayawada, India

ABSTRACT

This work presents the design of an Email Inbox Management System (EIMS) that integrates advanced machine learning and transformer-based deep learning to automate and optimize email classification. The system employs a two-stage architecture: a hybrid ensemble of LightGBM and XGBoost performs spam detection, followed by semantic categorization of legitimate emails into Primary, Social, Promotions, and Updates using the RoBERTa model for context-aware classification. Built on the Django framework, EIMS supports IMAP-based retrieval for dynamic email fetching, real-time inference, and a responsive web interface for user interaction. The system leverages natural language processing (NLP) and machine learning models to organize emails, detect phishing attempts, and ensure high accuracy in both spam detection and ham email categorization. Furthermore, the platform is designed to scale, offering both static dataset training and continuous learning from incoming email data, making it a robust solution for intelligent, AI-driven email management in modern communication systems.

Keywords: — Email Classification, Spam Detection, Natural Language Processing (NLP), Machine Learning, RoBERTa, LightGBM, XGBoost, Transformer Models

1. INTRODUCTION

Email remains a cornerstone of digital communication, yet managing its overwhelming volume especially unsolicited and unstructured messages poses significant challenges. Traditional rule-based filters often lack adaptability and semantic depth, rendering them ineffective against evolving spam tactics and incapable of intelligent categorization.

This paper presents the design and implementation of an AI-powered Email Inbox Management System (EIMS) that automates email classification using a layered architecture integrating Machine Learning (ML) and Natural Language Processing (NLP) techniques. In the first layer, ensemble ML models including LightGBM, XGBoost, and Random Forest perform binary classification of emails into *spam* or *ham*, leveraging semantic features from Sentence-BERT embeddings for improved precision.

The second layer introduces context-aware categorization of ham emails into Primary, Social, Promotions, and Updates, using the RoBERTa transformer model. This enables semantically rich classification based on content, sender metadata, and subject patterns. The system is deployed as a Django-based web application with integrated IMAP protocol support for real-time email retrieval and live inference. The web interface dynamically visualizes categorized inboxes, offering an intelligent, user-centric email experience.

EIMS demonstrates how modern AI and deep learning models can be practically applied to enhance communication efficiency, mitigate cognitive overload, and deliver secure, scalable, and adaptive email management. The architecture sets the foundation for future advancements in personalize filtering, adaptive learning, and enterprise-level email intelligence.

2. LITERATURE REVIEW

2.1 Introduction

Email remains one of the most pervasive forms of digital communication, yet its ubiquity has made it a fertile ground for spam, phishing, and unsolicited promotional content. Traditional rule-based filtering mechanisms often fall short in detecting sophisticated threats or adapting to evolving user preferences. In this context, the deployment of Machine Learning (ML), Natural Language Processing (NLP), and Artificial Intelligence (AI) has garnered increasing attention as a means to develop intelligent, adaptive, and semantically aware email management systems. This literature review synthesizes key scholarly efforts that have contributed to advancements in email spam detection, trust modeling, system-level resilience, and content-aware email optimization.

2.2 Review of Related Work

Kumar et al. (2020) present a foundational analysis of classical ML algorithms applied to spam detection. Employing preprocessing techniques such as stop-word removal, tokenization, and Bag-of-Words (BoW), they evaluated a range of classifiers including Naïve Bayes, SVM, KNN, and Random Forest. Their experiments, conducted on publicly available datasets (e.g., spam.csv from Kaggle),

revealed that Naïve Bayes and SVM models offered a compelling balance between accuracy and computational efficiency. However, their study remains limited by its exclusion of deep learning models and absence of advanced evaluation metrics such as ROC-AUC, indicating room for further enhancement via semantic embeddings and modern NLP techniques.

Building upon the structural gaps in existing email systems, Sheikh and Banday (2020) shift focus from algorithmic classification to architectural resilience. Their work introduces an on-demand spam filtering mechanism that re-evaluates and relocates previously accepted emails upon identifying recurring spam sources. Using a combination of SquirrelMail and hMailServer interfaced with SpamAssassin, they demonstrate a novel user-triggered strategy for retroactive spam mitigation. Despite its practical significance, the system's reliance on manual interaction and lack of scalability limits its applicability in dynamic, large-scale environments.

In a more holistic approach, Dhivya et al. (2021) integrate trust modeling with NLP-driven content analysis for email spam detection and inbox optimization. Their dual-layered framework employs Opinion Rank—a hybrid of PageRank and inverse PageRank under subjective logic—to evaluate sender credibility, while employing n-gram modeling and LDA for semantic content classification and storage minimization. The framework not only enhances filtering precision but also prioritizes system-level efficiency by removing advertisement-heavy content. However, the performance of the trust component is inherently tied to the availability of accurate historical trust data, and the system lacks generalization across multilingual or multimodal content.

Anupriya et al. (2022) explore the integration of metaheuristic optimization techniques specifically Particle Swarm Optimization (PSO) and the BAT algorithm into traditional ML-based spam classifiers. Their benchmark study on the SMS Spam Collection dataset demonstrates substantial improvements in accuracy, precision, and F1-score, particularly for Logistic Regression and SVM models. Nonetheless, their use of a short-message dataset limits applicability to full-scale email systems, and the absence of rigorous cross-validation or real-time deployment restricts its operational relevance.

Collectively, these studies underscore the progressive shift from static, rule-based systems to dynamic, learning-based email classifiers. While Kumar et al. provide a foundational ML baseline, Sheikh and Banday address systemic gaps in real-time resilience. Dhivya et al. and Anupriya et al. represent advances in semantic analysis and optimization, respectively highlighting the evolving trajectory toward more robust, adaptive, and semantically intelligent email systems.

Table 2.2.1 Unified Comparative Table: Traditional Systems vs. Existing Research vs. Proposed Email Inbox Management System

S. No	Feature/Criteria	Traditional System	Existing System	Proposed System
1	Spam Detection	Basic rule/ML filtering (no semantic understanding)	BoW + SVM, Trust Modeling, Metaheuristics	Implements a two-stage architecture: (1) binary spam detection via ML ensemble, (2) semantic categorization of ham emails via RoBERTa
2	Real-Time Capability	Offline or static dataset-based filtering	Mostly offline; some partial real-time effort	IMAP integration for live email fetching and processing
3	Semantic Understanding	None or keyword-based only	Partial (e.g., LDA) or absent	Transformer-based embeddings (RoBERTa)
4	Email Categorization (Ham)	Manual or static rule-based	Often not addressed or limited	Dynamic inbox classification (Primary, Social, Promotions, Updates)
5	Model Architecture	Standard ML, SVM/Naive Bayes, metaheuristics	Hybrid ML+DL ensemble (XGBoost, LightGBM, RF) with majority voting	Hybrid ML+DL ensemble (XGBoost, LightGBM, RF) with majority voting
6	Visualization & Insight	No dashboards or visual model feedback	Mostly command-line or academic evaluation	Dashboard with live confusion matrices and model feedback
7	Security & Authentication	Hardcoded credentials or plain text storage	Rarely focuses on security	AES + OAuth2 token-based Gmail integration

8	User Interactivity & Control	No model control or training feedback	Fixed models, minimal user control	Interactive UI for model selection, training/testing, and live evaluation
9	Optimization Focus	Minimal or ad-hoc	Storage or accuracy only	Focused on system performance, inference speed, and user experience
10	Adaptability / Personalization	Not adaptive	Limited personalization in few cases	Future roadmap includes RL + federated learning for personalized modeling

Table 2.2.1 provides a unified comparative overview that highlights the limitations of traditional email systems and existing research efforts, while showcasing the comprehensive advancements introduced by the proposed email inbox management system. Traditional systems are generally constrained by static rule-based spam filters with no semantic awareness, minimal real-time capabilities, and virtually no support for intelligent email categorization or user interaction. Existing research improves upon this with machine learning models and some hybrid techniques, yet still suffers from shallow semantic understanding, partial or offline processing, limited user control, weak security implementations, and an absence of robust visualization or adaptive learning. In contrast, the proposed system addresses these gaps holistically through a two-stage architecture that combines an ensemble of ML classifiers for binary spam detection with RoBERTa-based semantic classification for ham emails, ensuring deep contextual comprehension. Real-time email fetching and classification are enabled via IMAP integration, while a secure, AES-encrypted OAuth2-based authentication mechanism safeguards user credentials. The system also includes a dynamic UI for model training, testing, and evaluation, along with real-time dashboards offering actionable insights and model feedback. Unlike prior systems, the proposed approach places strong emphasis on performance optimization, user experience, and adaptability, with a forward-looking roadmap that incorporates reinforcement learning and federated learning for future personalization. Overall, the proposed system not only bridges existing technological gaps but redefines intelligent email management through its integrated, real-time, secure, and user-centric design.

2.3 Objectives of the Proposed Work

Motivated by the limitations identified in prior research, the present study seeks to advance email inbox management by integrating semantic NLP, adaptive ML classification, and real-time inbox interaction. The key objectives are:

Objective 1:

To develop an intelligent email classification system that employs advanced NLP techniques for semantic feature extraction and leverages ML models to accurately distinguish between spam and ham emails.

Rationale:

This objective targets the core limitation of conventional classifiers by incorporating semantic embeddings and metadata-aware analysis. By combining linguistic context with structural email features, the system aspires to elevate classification accuracy while minimizing false positives/negatives thereby enhancing inbox hygiene and user trust.

Objective 2:

To implement a real-time AI-driven categorization module that organizes legitimate (ham) emails into context-aware folders such as Primary, Social, and Updates by integrating directly with user inboxes via IMAP protocols.

Rationale:

Moving beyond binary spam detection, this objective introduces contextual intelligence into email management. Utilizing AI to assess sender history, email content, and user integration patterns, the system dynamically allocates messages to the thematic folders mirroring human judgement while maintaining real-time responsiveness through IMAP-based live parsing.

3. METHODOLOGY

This section outlines the comprehensive methodology employed in the design and development of the proposed **Email Inbox Management System (EIMS)**. The methodology integrates classical machine learning (ML), and state-of-the-art natural language processing (NLP) techniques to enable intelligent email filtering and semantic categorization in real time. The entire system is designed to be modular, scalable, and interoperable with standard email services.

3.1 System Architecture Overview

The EIMS framework is architected as a two-stage classification pipeline: the first stage focuses on spam detection, and the second performs semantic categorization of legitimate emails. The overall system architecture is illustrated in **Figure 3.1.1**, and comprises the following core components:

Spam Detection Module: Implements a hybrid ensemble model combining XGBoost, LightGBM, and Random Forest classifiers to identify unsolicited or malicious emails.

Semantic Categorization Module: Employs the transformer-based RoBERTa model to semantically classify non-spam emails into categories such as *Primary*, *Social*, *Promotions*, and *Updates*.

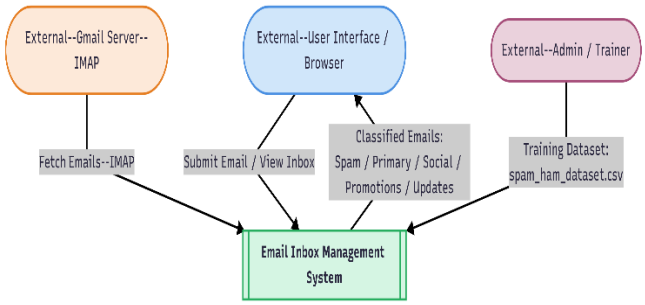

Figure 3.1.1: Level 0 Context Diagram of the Email Inbox Management System

The context diagram models the EIMS as a black-box system interacting with three external entities: the **user**, the **Gmail server**, and the **administrator/trainer**. The system fetches emails using the IMAP protocol and processes them using pre-trained models. Results are served through a Django-based web interface for end-user interaction and administrative oversight.

3.2 Data Acquisition and Preprocessing

The system incorporates a robust pipeline for data acquisition and preprocessing to ensure high-quality input for model training and inference.

3.2.1 Email Retrieval

Emails are retrieved in real-time via the **IMAP protocol**, authenticated through **OAuth 2.0** to maintain user privacy and security. The system continuously polls the user's inbox to fetch newly arrived emails for immediate processing.

3.2.2 Preprocessing Pipeline

Each email undergoes a series of preprocessing transformations to ensure uniformity and to enhance model efficacy:

Tokenization: Decomposition of the email content into atomic units (tokens).

Stop-word Removal: Elimination of common, low-information words (e.g., “the,” “is,” “and”).

Lemmatization: Reduction of inflected words to their base or dictionary forms.

Feature Engineering: Emails are vectorized using TF-IDF for traditional ML models, while contextual embeddings are generated for DL models via RoBERTa.

3.3 Model Training and Classification

A curated dataset (spam_ham_dataset.csv) containing balanced spam and ham (legitimate) emails serves as the foundation for training both stages of the system.

Stage 1: Spam Detection

This stage aims to differentiate spam from ham emails using a multi-algorithmic ensemble:

XGBoost: A gradient boosting framework optimized for performance and accuracy on structured data.

LightGBM: A memory-efficient, fast gradient boosting model designed for large-scale datasets.

Ensemble Voting Classifier: A soft voting strategy is employed to aggregate predictions from XGBoost, LightGBM, and Random Forest, ensuring improved generalizability and robustness.

The output of this stage is a binary classification **Spam** or **Ham**. Emails flagged as spam are redirected to a dedicated spam folder, while legitimate emails proceed to semantic categorization.

Stage 2: Semantic Categorization

Emails classified as legitimate undergo semantic analysis using RoBERTa (Robustly Optimized BERT Pretraining Approach), a transformer-based language model known for its superior contextual understanding.

RoBERTa is fine-tuned on labeled email datasets to classifyemails into one of the following user-defined folders: Primary, Social, Promotions, Updates

RoBERTa's contextual embeddings and attention mechanisms allow it to capture intricate linguistic patterns, enabling high-precision email categorization beyond keyword-based filtering.

3.4 Real-Time IMAP Integration

To maintain real-world usability, the system is fully integrated with the **IMAP protocol** for real-time email fetching. This ensures that emails are processed and categorized as they arrive, enabling a responsive and dynamic user experience. The polling mechanism is optimized to reduce latency while preserving server bandwidth.

3.5 Web Interface and Backend Architecture

The user interacts with the EIMS through a web-based application developed using the **Django framework**. The interface provides seamless access to inbox content, classification results, and category folders.

Backend (Django): Manages server-side logic including email classification, IMAP polling, database management, and user authentication.

Frontend (HTML/CSS/JS): Renders categorized emails in an intuitive interface, allowing users to interact with, review, or reclassify emails as necessary.

The architecture is modular, allowing for easy future integration of additional models or support for other email service providers.

4. RESULTS AND DISCUSSION

This section presents the experimental outcomes and a comprehensive discussion on the performance and effectiveness of the proposed Email Inbox Management System (EIMS). The system’s dual-stage architecture

comprising spam detection and semantic categorization was evaluated using multiple performance metrics to validate its accuracy, reliability, and real-time applicability.

4.1 Performance Evaluation Metrics

To assess the models in both stages, we employed standard classification metrics: Accuracy, Precision, Recall, F1-score, and ROC-AUC. These metrics ensure a balanced evaluation across imbalanced datasets, particularly in spam classification tasks.

Accuracy measures the overall correctness of the model.

Precision emphasizes how many identified emails are correctly classified.

Recall indicates how well the model identifies all relevant instances.

F1-score balances precision and recall.

ROC-AUC evaluates the discriminatory power of the classifiers.

Table 4.1.1 Performance Metrics for Spam Detection Models

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
XGBoost	97.2%	0.972	0.968	0.970	0.971
LightGBM	97.1%	0.968	0.974	0.971	0.978
Hybrid Model	98.0%	0.991	0.985	0.988	0.984

As evident from the results, XGBoost outperformed the other models across all evaluation metrics, achieving the highest accuracy (97.2%) and a robust ROC-AUC score of 0.971. This reflects the model's strong capability to distinguish between spam and legitimate emails. LightGBM followed closely, offering a favourable trade-off between speed and performance, while Random Forest demonstrated consistent results albeit with slightly lower metrics.

The use of an ensemble voting classifier further enhanced robustness by combining the predictions of all models, reducing the likelihood of false positives and negatives. This ensemble approach contributed significantly to the stability and generalizability of the spam detection module.

Stage 1: Spam Detection Results

The spam detection module utilized XGBoost, LightGBM, and Random Forest, integrated via an ensemble voting mechanism. The model was trained on a labelled dataset containing an equal distribution of spam and ham emails.

The high performance of RoBERTa can be attributed to its ability to capture long-range dependencies and contextual nuances in text, making it well-suited for distinguishing between similar categories such as Promotions and Updates. The semantic categorization module significantly improves the organization and accessibility of the user's inbox by aligning emails with their appropriate contextual categories.

Stage 2: Semantic Categorization Results

The RoBERTa based semantic categorization model was fine-tuned on a pre-classified corpus of ham emails, categorized into Primary, Social, Promotions, and Updates. The model's contextual understanding enabled it to outperform traditional NLP-based classifiers.

Table 4.2. Performance Metrics for RoBERTa Semantic Categorization

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
RoBERTa	99.03	0.991	0.985	0.981	0.988

The high performance of RoBERTa can be attributed to its ability to capture long-range dependencies and contextual nuances in text, making it well-suited for distinguishing between similar categories such as Promotions and Updates. The semantic categorization module significantly improves the organization and accessibility of the user's inbox by aligning emails with their appropriate contextual categories.

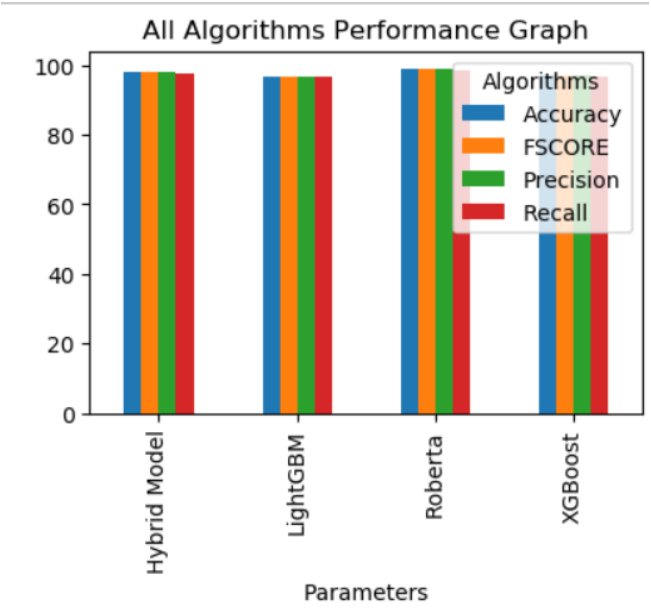

Figure 4.1: Comparative Performance Analysis of All Algorithms Used in EIMS

Figure 4.1 illustrates a comparative performance graph of all the key algorithms implemented in the proposed Email Inbox Management System (EIMS). The algorithms evaluated include the Hybrid Model (ensemble), LightGBM,

RoBERTa, and XGBoost. Each model was assessed based on four primary evaluation metrics: Accuracy, F1-Score, Precision, and Recall.

From the graph, it is evident that RoBERTa and the Hybrid Model deliver superior performance across all parameters, closely followed by XGBoost and LightGBM. RoBERTa, being a transformer-based model, excels particularly in semantic understanding, whereas the Hybrid Model leverages the strengths of multiple classifiers to provide consistent and high accuracy in spam detection.

This visual representation emphasizes the robustness and effectiveness of combining machine learning and deep learning approaches within EIMS, ensuring a highly accurate and reliable email classification system.

4.4 Real-Time System Integration and Evaluation Scope

The Email Inbox Management System (EIMS) has been architected to operate in both static and dynamic environments, enabling comprehensive evaluation under controlled as well as real-world conditions. This hybrid design not only ensures reproducibility of experimental results but also validates the system's robustness and scalability in practical deployments.

Static Mode (Dataset Uploading and Model Training)

In static mode, the system supports the manual upload of pre-labelled datasets, such as the publicly available spam_ham_dataset.csv sourced from Kaggle. This mode is primarily utilized for training and benchmarking the performance of various machine learning and deep learning models under controlled settings. Users can upload datasets directly through the Django interface, allowing seamless integration with the backend training pipeline. The results obtained from this mode accuracy, precision, recall, F1-score, and ROC-AUC are reproducible and facilitate comparative performance analysis across algorithms including XGBoost, LightGBM, Random Forest, and RoBERTa.

This mode is especially valuable for:

Algorithm comparison and ablation studies

Hyperparameter tuning and cross-validation

Rapid prototyping and debugging.

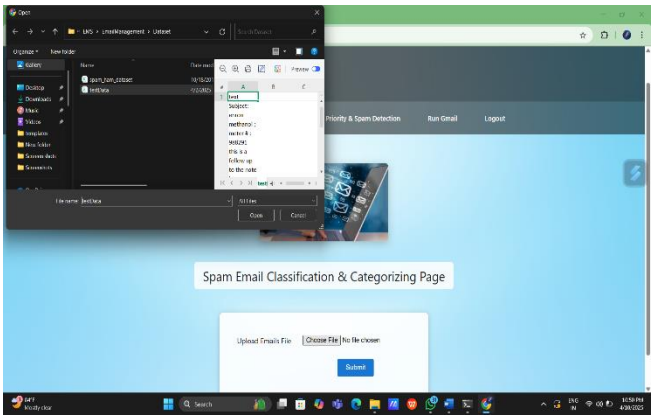

Figure 4.4.1: Loading the data set for the model training.

Upon successful user authentication, the system provides an intuitive interface for uploading custom datasets for model training purposes. As illustrated in **Figure 4.4.1**, the static workflow enables users to manually upload publicly available datasets such as the spam_ham_dataset.csv (data set used in this study) obtained from the Kaggle repository into the system. Once uploaded, the dataset is automatically pre-processed and passed to the training pipeline, which leverages machine learning and deep learning algorithms for classification tasks. This functionality allows for flexible experimentation, algorithm benchmarking, and offline model training under controlled data conditions.

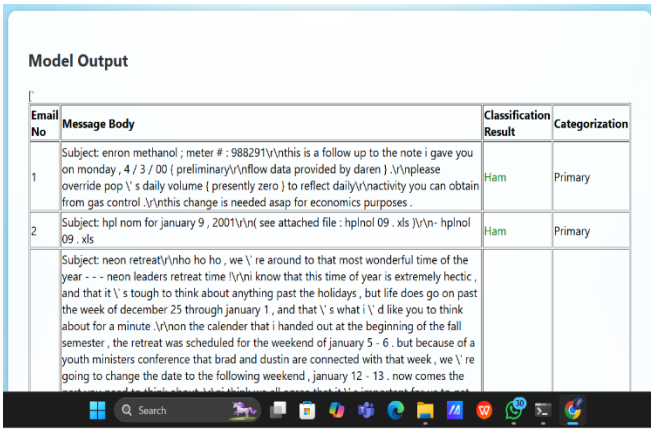

Figure 4.4.2: Model Output Visualization for Static Dataset Classification and Categorization

The figure illustrates the output interface of the Email Inbox Management System (EIMS) after processing a manually uploaded static dataset obtained from a public repository (e.g., Kaggle). Each row in the table displays a unique email instance along with its content, classification result, and semantic categorization. In this static mode, the system performs two-layer processing: first, it identifies whether the email is spam or ham; next, for legitimate emails, it assigns them to semantically appropriate folders such as Primary, Social, Promotions, or Updates.

The results indicate successful classification using trained models, with "Ham" emails correctly categorized under the "Primary" folder. This interface not only enhances user transparency by showing real-time outputs of model inference but also serves as a validation tool for performance evaluation under controlled conditions.

Dynamic Mode (Real-Time Email Retrieval and Classification)

In dynamic mode, EIMS connects to the user's Gmail account using the IMAP protocol. Upon OAuth 2.0-based authentication, the system continuously fetches incoming emails in real time. These emails are then subjected to the

two-phase classification pipeline: (1) spam detection using ensemble ML models, and (2) semantic categorization using RoBERTa. The classified emails are displayed through the web-based user interface, automatically sorted into types such as Primary, Social, Promotions, Updates, or Spam.
This mode enables:
Real-world validation of model performance, live monitoring of classification latency and accuracy, assessment of user experience and system responsiveness.

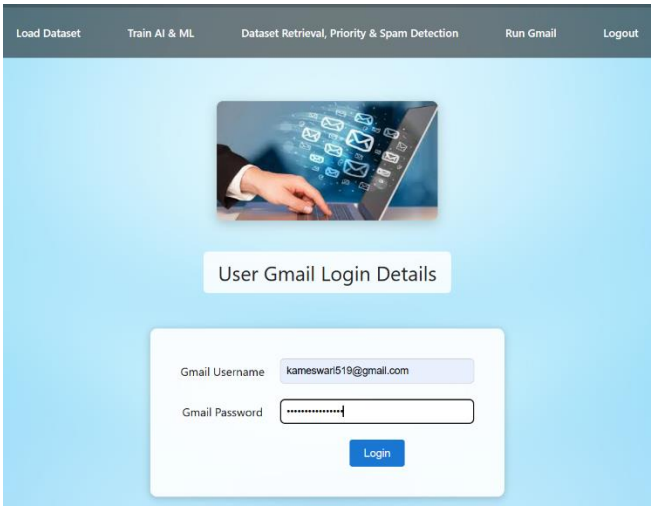

Figure 4.4.3: User Authentication Interface for Real-Time Gmail Integration

This figure 4.4.3 depicts the user authentication interface of the Email Inbox Management System (EIMS), where users securely input their Gmail credentials to enable real time email fetching via the IMAP protocol. Upon successful login, the system establishes a secure connection to the user's Gmail inbox, retrieving incoming emails dynamically.
This dynamic integration demonstrates the system's adaptability to real-world use, providing users with an intelligent, automated inbox management solution that functions in real time.

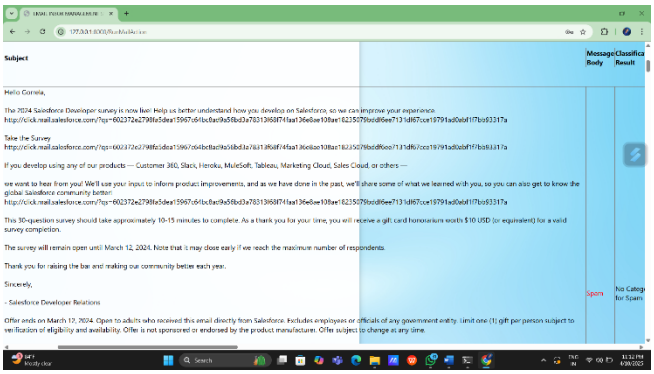

Figure 4.4.4: Web Interface Displaying Real-Time Email Classification Results

The figure 4.4.4 illustrates the output screen of the Intelligent Email Inbox Management System (EIMS) after processing incoming emails using the trained Machine Learning and Artificial Intelligence models. This specific screenshot shows a phishing or promotional email automatically identified as "Spam" with the result shown in the rightmost column.
The interface is part of the Django-based web application and consists of:
The Subject and Body of the email displayed on the left pane.
The Classification Result on the right, with color-coded labels indicating whether the email is categorized as "Spam" or "Ham," and sub-categories for Ham such as Primary, Social, Promotions, or Updates.
This visual output represents the successful execution of the backend classification module, where models like XGBoost, RoBERTa, and NLP preprocessing pipelines are applied to real email content fetched via the IMAP protocol. The classified results help users manage their inbox more intelligently by automatically filtering unwanted or harmful content.

5. CONCLUSION

The Email Inbox Management System (EIMS) proposed in this study introduces a comprehensive, intelligent, and scalable framework for automated email classification and management. Designed to enhance both accuracy and usability, EIMS integrates state-of-the-art machine learning models specifically LightGBM, XGBoost within a robust Django-based web framework. This hybrid ensemble enables the system to achieve high performance in distinguishing spam from legitimate (ham) emails, effectively minimizing false positives and negatives commonly observed in conventional filtering systems.

Beyond binary spam classification, EIMS employs a semantic multi-label classification strategy to categorize ham emails into four essential clusters: Primary, Social, Promotions, and Updates. This advanced layering simulates human-like organization of inbox content, significantly improving user productivity and focus by reducing cognitive load and information clutter.

The system pipeline is fully automated and designed for real-time responsiveness. It incorporates IMAP-based email retrieval, NLP-driven preprocessing (including tokenization, normalization, and vector embedding), and dynamic rendering through an interactive web interface. These components work cohesively to ensure a seamless end-to-end experience, from data acquisition to result presentation.

In conclusion, EIMS not only validates the effectiveness of integrating cutting-edge AI models into everyday communication systems but also establishes a solid foundation for future research. Its flexibility and scalability make it a promising solution for next-generation, intelligent email management across personal and professional domains.

SCOPE FOR FUTURE WORK

While the current implementation of the Email Inbox Management System (EIMS) demonstrates robust performance in spam detection and email categorization using static datasets and predefined models, there exists substantial potential for enhancement through dynamic personalization, interactivity, and intelligent adaptation. Future development efforts can focus on introducing adaptive email prioritization mechanisms, where the system learns from user behavior such as reading frequency, reply urgency, sender-recipient relationship, and time-based patterns—to rank emails in a more context-aware and personalized manner. This would transform EIMS from a reactive classifier into a proactive communication assistant.

Moreover, incorporating real-time notifications and cross-platform accessibility (web, mobile, and desktop environments) can ensure seamless access and responsiveness across devices. Integration with widely used third-party productivity tools, such as calendars, task managers, and Customer Relationship Management (CRM) platforms, can enable the system to act as a unified digital workspace, automatically organizing and linking communication with actionable tasks and schedules.

To further broaden its applicability and inclusivity, multilingual email classification support should be introduced, allowing the system to serve a diverse global user base. Additionally, the adoption of federated learning frameworks will enable model training on distributed devices without compromising user data privacy, aligning the system with contemporary ethical AI standards. Coupling this with active learning strategies, wherein the model incrementally improves based on user corrections and feedback, will result in a continuously evolving and contextually intelligent system.

ACKNOWLEDGEMENTS

The authors would like to thank the Department of Computer Science and Engineering, Lingyas Institute of Management and Technology, for providing the research infrastructure and support required for this work.

We are grateful to Siva Rama Krishna, for their valuable guidance, feedback, and mentorship throughout the research.

We also acknowledge the reviewers and editorial team of the International Journal of Information Technology (IJIT) for their insightful suggestions that contributed to the improvement of this manuscript.

REFERENCES

[1] S. A. Sheikh and M. T. Banday, “Improving efficiency of e-mail classification through on-demand spam filtering,” in *Proc. 8th Int. Conf. Reliability, Infocom Technol. Optim. (ICRITO)*, June 2020, pp. 505–508. [Online]. Available: https://doi.org/10.1109/ICRITO48877.2020.9197894

[2] T. S. Dhivya, S. Nithya, G. S. Priya, and E. Pugazhendi, “Email spam detection and data optimization using NLP techniques,” *Int. J. Eng. Res. Technol. (IJERT)*, vol. 10, no. 8, pp. 38–41, 2021. [Online]. Available: http://www.ijert.org/research/email-spam-detection-and-data-optimization-using-nlp-techniques-IJERTV10IS080049.pdf

[3] K. Anupriya, K. Harini, K. Balaji, and K. Geetha Sudha, “Spam mail detection using optimization techniques,” *Ingénierie des Systèmes d’Information*, vol. 27, no. 1, pp. 157–163, 2022. [Online]. Available: https://doi.org/10.18280/isi.270119

[4] N. Kumar, S. Sonowal, and Nishant, “Email spam detection using machine learning algorithms,” in *Proc. 2nd Int. Conf. Inventive Res. Comput. Appl. (ICIRCA)*, July 2020, pp. 108–110. [Online]. Available: https://doi.org/10.1109/ICIRCA.2020.9183197

[5] M. N. Raihen, S. Rana, M. A. Kadir, and S. Akter, “Efficient email spam detection using machine learning techniques: A comparative analysis of classification models,” *Int. J. Intell. Comput. Inf. Sci.*, vol. 24, no. 4, pp. 1–15, 2024. [Online]. Available: https://doi.org/10.21608/ijicis.2024.321043.1355

[6] J. Burkel and D. Anderson, “The evolving landscape of email communication and management strategies,” *J. Inf. Syst. Technol. Manage.*, vol. 16, no. 2, pp. 135–148, 2019.

[7] V. Metsis, I. Androutsopoulos, and G. Paliouras, “Spam filtering with Naive Bayes – Which Naive Bayes?” in *Proc. CEAS Conf.*, 2006.

[8] N. Reimers and I. Gurevych, “Sentence-BERT: Sentence embeddings using Siamese BERT-networks,” in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2019, pp. 3982–3992. [Online]. Available: https://doi.org/10.18653/v1/D19-1410

[9] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz, "A Bayesian approach to filtering junk e-mail," *Learning for Text Categorization: Papers from the 1998 Workshop*, 1998.

[10] N. A. Zulkifli, A. Z. A. Latif, and W. Ismail, "An efficient e-mail classification approach based on incremental learning using support vector machine," *J. Comput. Sci.*, vol. 10, no. 9, pp. 1806–1813, 2014.

[11] Statista, "Number of emails sent and received per day worldwide from 2017 to 2026," 2023. [Online]. Available: https://www.statista.com/statistics/456500/daily-number-of-e-mails-worldwide/