iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** III **Month of publication:** March 2026

DOI: https://doi.org/10.22214/ijraset.2026.78656

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue III Mar 2026- Available at www.ijraset.com

Smart Email Sorting: A Machine Learning Approach for Enhanced Productivity

Sesetti Yaswini Priya[1], Bala Indra Sena Redddy [2], Rajulapati Uday Malleswar[3], Bulla Vijay Kumar[4], CH. Jeevana Priya[5]
[1, 2, 3, 4]*UG-Computer Science and Engineering, SRK Institute of Technolog, Vijayawada, Andhra Pradesh*
[5]*Professor, Computer Science and Engineering, SRK Institute of Technology, Vijayawada, Andhra Pradesh*

Abstract: ***Smart Email Sorting System uses Natural Language Processing (NLP) and a fine-tuned DistilBERT model to classify emails, by analyzing the contents of an email message and determining if they fall into categories (e.g., Invoices, Meetings, Project Update) or not. It also pulls deadlines and key info from unstructured text. Using OAuth integration with Gmail, the system sorts emails according to urgency and stresses time-sensitive engagements, allowing users to swiftly locate significant communications in an increasingly digital environment where they can manage their inboxes more efficiently. The system eliminates the manual overhead on a massive scale by automating email sorting and prioritization based on predictive capabilities. It prevents essential emails from getting lost in the shuffle and reminds users to respond quickly to significant activities. This results in increased productivity, efficient time management and more streamlined workflow; offering a solution to managing copious amounts of correspondence both professionally and personally.***

Keywords: Smart Email Sorting, Natural Language Processing (NLP), DistilBERT, Email Classification, Deadline Extraction, Machine Learning, Deep Learning, Text Classification, Email Automation.

.

I. INTRODUCTION

In modernized quick digital workspace, email is still one of best personal and professional form of communication. But the sheer volume of daily messages is overwhelming. Classic email systems sort chronologically and use rule-based filters that miss context, sentiment or urgency. This leads to professionals spending more time organizing their inboxes, forgetting deadlines and crucial tasks along the way which results in a loss of productivity and unorganized flowing task management.

In order to overcome these issues, Smart Email Sorting System is presented under the mode of AI which helps in managing emails effectively. It uses sophisticated machine learning approaches to automatically classify and prioritize emails. Instead of regular keyword-based solutions, context aware models are used to detect what users want in messages and provides information accordingly. This addresses the issue of important emails being buried away without users having to be constantly proactive, alleviating up to 80% of workload allowing them to deal actually meaningfully with their emails.

Dynamics Of The System: At the centre of this system, we have a fine-tuned DistilBERT model and an NLP-based deadline extraction engine. It fetches, and processes emails using a secure Gmail OAuth integration which categorizes them into invoice types, meetings, HR and legal. It picks out key dates, calculates how much time remains as of now and highlights messages that are most urgent. React and FastAPI based, it has a scalable, secure architecture and role-based dashboards that increase productivity and reduce missed deadlines.

II. RELATED WORK

1) Hybrid DistilBERT + NLP Extraction for Email Classification: Emails are processed through an efficient DistilBERT transformer model in our system for a real-time prediction and classification. It pairs deep contextual understanding with an NLP-powered extraction engine that pulls out crucial details such as dates to extract deadlines and remaining time. It effectively manages complex email content through a combination of context-aware classification and accurate extraction logic. This provides better performance than more rudimentary keyword/rule methods, which can fail to capture nuanced communication and subtle urgency in work email contexts

2) Traditional Sorting and Rule-Based Limitations: Many conventional systems of email sorting depend upon rules defined earlier that associate keywords or sender info to folders. They are great for basic spam filtering or keeping newsletters tidy, but do not cope well with complicated emails that have multiple discussion points/action items. These systems do not measure urgency based on context. Moreover, their lack of intelligence and context-aware features leads to limited dynamic prioritization and decision-making based on pre-defined logic that does not necessarily fit modern productivity needs.

3) Modern Email Clients and Semi-Automated Systems: Some modern email clients add to inbox management functionality with tools such as smart replies and rudimentary priority tagging that help surface important messages. While these solutions allow users to process emails more quickly, they still base themselves either on generic algorithms or manual flagging methods that limit personalization. Automation, like promotions or social tabs, provides me basic organization but not deeper intelligence. These systems are not capable of good task extraction or deadline tracking, so they fall short for email management and advanced productivity.

4) Scalable AI-Powered Email Management: FastAPI, React and other powerful databases are used to handle many users and high email volume. They also allow for safe email synchronization and storage with secure multi-user isolation and Gmail OAuth integration. Such systems store structured, queryable data for real-time dashboard analytics. They do super well in workspaces which is what allows for fast prioritization and accurate tracking of deadlines and showcases that building a secure, AI-based lighter productivity system to solve modern communication needs is very much feasible.

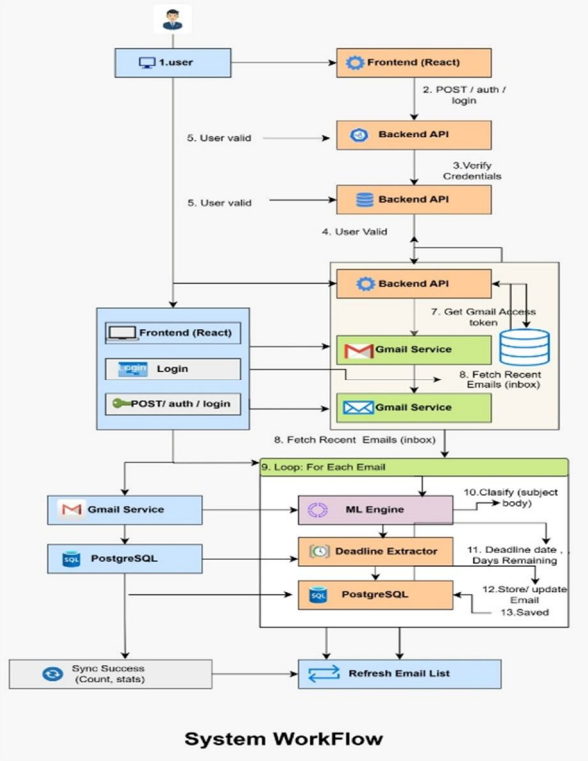
III. PROBLEM STATEMENT

Creativity is attracted to chaos, and traditional email systems are a cesspool: they have a strict chronological sorting of messages that deliver information overloads several times a day, most likely with missed deadlines.To overcome this Smart Email Sorting System take the approach of using a pre-trained DistilBERT model for context-aware classification and AI to prioritize emails based on its content, and determine the deadlines associated with emails. It uses a clever algorithm to let you know with days remaining, and flags up emails written in an urgent tone of voice; is robust enough for secure multi-user isolation; provides analytics dashboards, hungrily keeping inbox clutter down (and productivity up).

IV. WORKFLOW

The following flowchart helps us visualize Smart Email Sorting System workflow starting with user authentication using React web frontend and backend api. Once you log in, the system connects to Gmail using OAuth and downloads your emails. It provides safe access to the incoming messages, synchronized with the application environment without blocking it.

It then handles every incoming email through an ML engine for the classification and it passes a deadline extractor to be reported its important dates and urgency. The processed data is stored in PostgreSQL and updated per usage dashboards. This keeps emails organized, helps users gain insights in real time as well as priorities and improve their productivity.



System WorkFlow

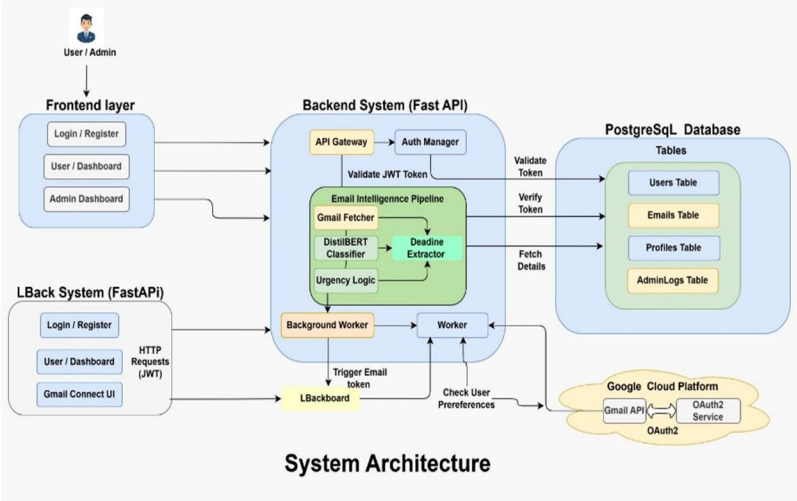
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue III Mar 2026- Available at www.ijraset.com

V. PROPOSED SYSTEM

To address these limitations, this project proposes Smart Email Sorting, an intelligent email classification and prioritization system powered by machine learning and natural language processing. The proposed system automatically classifies incoming emails into 17 business-specific categories such as Invoices, Orders, Customer Support, Legal, and Urgent, enabling users to immediately understand the nature of each email without reading its full content. In addition, the system extracts deadlines and time-critical information to highlight urgent messages. Unlike existing email applications, the proposed system is designed specifically for non-technical business users, using a simplified dashboard with icon-based navigation, clear visual cues, and no technical jargon. An admin interface allows manual correction of misclassified emails, and this feedback is used to continuously retrain and improve the model.

VI. SYSTEM ARCHITECTURE

It is designed using an asynchronous microservice-based system to be scalable and operate efficiently. A modern React dashboard serves as the frontend, making it very interactive and engaging for users, while FastAPI is used to manage backend services. The async workflow of tasks run with Celery and executed through Redis. Text analysis and classification using machine learning models such as BERT, SpaCy. PostgreSQL is a powerful and reliable database layer that addresses the need for organized data storage, consistency, and efficient retrieval throughout your application.



VII. MODULE DESCRIPTION

A. User Module

Manages secure personal accounts, Gmail synchronization, and isolated data access so users only interact with their own emails.
Inputs: Login credentials, Google OAuth consent, and "Sync Emails" button triggers.
Outputs: JWT session tokens, personalized dashboard alerts (deadlines/overdue tasks), and a categorized email table.

B. Admin Module

Provides system-wide oversight using Role-Based Access Control (RBAC), allowing admins to monitor overall performance without accessing private user data.
Inputs: Admin login credentials and dashboard navigation.
Outputs: Global analytics, key metrics (total vs. urgent emails processed), and category distribution visualizations.

C. System Module (Core Pipeline)

The backend engine that runs invisibly during syncs, bridging the Gmail API, AI models, and the database.
Inputs: Raw email data (Subject and Body) fetched from the Gmail API.
Outputs: AI category and confidence scores (DistilBERT), extracted deadlines, auto-calculated urgency flags ($\leq$ 2 days), and enriched database records mapped securely to the user.

VIII. DATASET DISCRIPTION

The data set includes actual emails from portals like Enron corpus as well as synthetically generated data so that all communication scenarios are covered. Key attributes include subject, body, sender domain and date received for each record. Such features produce the context meaning and time information necessary for successful email classification and deadline retrieval. This well-defined dataset allows the model to learn and adapt from diverse writing styles, formats, and business scenarios; thus enhancing its capability to generalize across different forms of professional and personal emails.

Dataset of all your organization's data is categorized into invoices, meetings HR, project-related work & general communication. Data - Preprocessing data cleaning, anonymization, tokenisation/Label Encoding. Then the data is split in a 70:15:15 ratio into train, validation and test sets. This allows to ensure model training, tuning and evaluation that prevent overfitting while increasing generalization, accuracy or reliability of email classification and deadline detection in a real world usage.

IX. ALOGORITHM USED

Smart Email Sort System uses a hybrid algorithmic approach consisting of deep learning, Natural Language Processing (NLP) and heuristic methods. By integrating this data with the documents, critical events with timeframes are extracted from emails and logged accurately so that the email threads are prioritized effectively and tasks automated thereby improving performance level with decreased resources utilization.

1) DistilBERT (Transformer-BasedModel): DistilBERT is a lightweight version of BERT used for email classification. It uses self-attention to understand full email context (subject + body) and captures semantic meaning. The model is fine-tuned to classify emails into categories like invoices, meetings, and HR, providing accurate predictions with confidence scores.

2) TF-IDF& Statistical Classification: TF-IDF converts email text into numerical vectors based on word importance. These vectors are passed to classifiers like Naive Bayes or SVM for fast predictions. It acts as a baseline model, offering quick and efficient classification when deep learning models are not required.

3) Named Entity Recognition (NER) – spaCy: NER identifies important entities like dates and times from email text. Using spaCy, the system detects temporal expressions such as “March 15” or “tomorrow,” enabling accurate deadline extraction without relying only on fixed patterns.

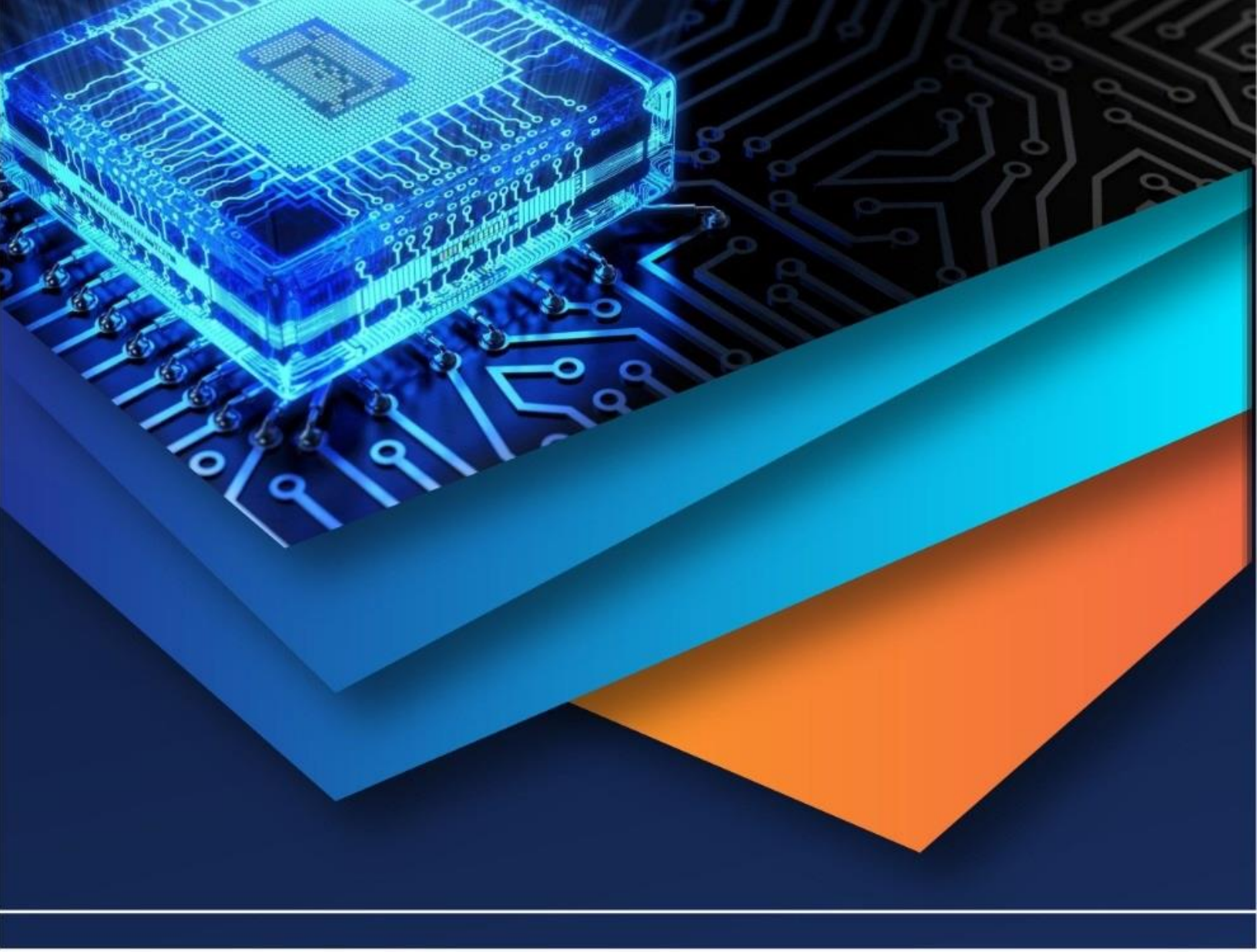
4) Heuristic Rule-Based Algorithm: This logic calculates remaining days from extracted dates and determines urgency. Emails are marked urgent if deadlines are near or urgency keywords are present. Additional rules prevent misclassification, ensuring accurate prioritization and better decision-making.

X. CONCLUSION AND FUTURE WORK

This innovation illustrates a significant gain in productivity by embedding an intelligent machine learning filter, rather than traditional rule-based filters such as spam or data loss filters. The model is fine-tuned DistilBERT and refined using NLP for correct classification of emails and deadline extraction. This system does more than store information; it actively flags urgent tasks to minimize missed communications and streamline operations. This leads to privacy along with usability through role-based dashboards, reducing excess cost and improving general decision making throughout email processing via secure FastAPI architecture, new GMAIL OAuth. Future developments involve adding sentiment analysis to help prioritize findings, asynchronous processing with Celery and Redis for better performance, as well as automatic OAuth token refresh. Features like real-time notifications using WebSockets and webhook systems would enable a better response. The Lightning Web Compoents will provide enterprise-grade analytics with Power BI integration and docking-based deployment for scalability, insights, and management in large-scale environments.

REFERENCES

[1] Vaswani et al., “Attention Is All You Need,” in Advances in Neural Information Processing Systems (NeurIPS), 2017.
[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” in Proc. NAACL-HLT, 2019.
[3] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, “DistilBERT, a Distilled Version of BERT,” arXiv preprint arXiv:1910.01108, 2019.
[4] M. J. Eppler and J. Mengis, “The Concept of Information Overload: A Review of Literature,” The Information Society, vol. 20, no. 5, pp. 325–344, 2004.
[5] T. Chen et al., “MLOps Best Practices, Challenges and Maturity Models,” Journal of Software Engineering, 2025.
[6] R. Kumar et al., “A Comparative Evaluation of a Multimodal Approach for Unwanted Email Detection,” Electronics, MDPI, 2025.
[7] P. Singh et al., “BERT-Based Phishing Email Classification,” in Proc. COMPSAC Workshop, 2023.
[8] A. Alshamrani et al., “EmailNet: Efficient Email Classification Based on Graph Similarity,” in Proc. INASS Conference, 2023.
[9] J. Li et al., “A Survey on Named Entity Recognition,” Journal of Natural Language Processing, 2023.
[10] M. Honnibal and I. Montani, “spaCy 2: Natural Language Understanding,” 2017.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY