Email Triage Assistant

AI-Powered Email Management System

---

1. Executive Summary

The **Email Triage Assistant** is an intelligent system designed to optimize email workflows by leveraging advanced Natural Language Processing (NLP) and Large Language Models (LLMs). It automates the process of reading, organizing, prioritizing, and responding to emails.

This solution addresses the growing challenge of email overload by enabling users to quickly understand, prioritize, and act on incoming messages with minimal manual effort.

---

2. Problem Statement

Modern professionals receive a high volume of emails daily, leading to:

- Time-consuming manual sorting and reading
- Missed or delayed responses to important emails
- Cognitive overload from long email threads
- Inefficient communication workflows

There is a need for a system that can intelligently manage emails while preserving context and accuracy.

---

3. Proposed Solution

The Email Triage Assistant provides an end-to-end automation pipeline that:

- Summarizes email threads into concise insights

- Classifies emails by intent and urgency
- Assigns priority levels dynamically
- Generates context-aware response suggestions

The system integrates seamlessly with existing email platforms and enhances productivity without disrupting user workflows.

---

4. Key Features

4.1 Intelligent Email Summarization

- Condenses long email threads into structured summaries
- Extracts:
    - Key discussion points
    - Decisions made
    - Action items
    - Deadlines

Outcome: Reduces reading time significantly

4.2 Automated Email Classification

Emails are categorized into meaningful classes such as:

- Urgent
- Action Required
- Informational

- Meeting / Scheduling
- Low Priority / Spam

Approach:

- Supervised NLP classification
- Context-aware embeddings

4.3 Dynamic Priority Assignment

Each email is assigned a priority level:

- High
- Medium
- Low

Factors Considered:

- Sender importance
- Time sensitivity
- Keywords indicating urgency
- Historical interaction patterns

4.4 Smart Response Generation

- Generates context-aware replies using LLMs
- Supports multiple tones:

    - Formal
    - Casual
    - Concise

Benefit: Reduces effort in drafting repetitive emails

4.5 Context Awareness

- Maintains thread-level understanding
- Avoids redundant or irrelevant responses
- Ensures continuity in communication

---

5. System Architecture

5.1 High-Level Design

Email Provider (Gmail / Outlook)
↓
API Integration Layer
↓
Preprocessing Engine
↓
AI Processing Modules
├── Summarization Engine
├── Classification Engine
├── Priority Scoring System
└── Response Generator
↓
Database / Storage
↓
User Interface (Dashboard)

5.2 Component Breakdown

5.2.1 Email Integration Layer

- Connects via secure OAuth2
- Fetches:
    - Email content
    - Metadata
    - Thread history

5.2.2 Preprocessing Engine

- Cleans HTML and signatures
- Reconstructs email threads
- Performs tokenization and normalization

5.2.3 AI Processing Layer

Summarization Module

- Transformer-based models
- Hybrid extractive + abstractive approach

Classification Module

- Multi-class text classifier
- Fine-tuned on email datasets

Priority Engine

- Hybrid rule-based + ML scoring

- Example scoring formula:

  Priority Score = Sender Weight + Keyword Score + Deadline Urgency

Response Generator

- Prompt-engineered LLM
- Controlled output formatting

5.2.4 Storage Layer

- Stores processed data:
  - Email summaries
  - Categories
  - Priority levels
- Technologies:
  - PostgreSQL / MongoDB

5.2.5 User Interface

- Clean dashboard for:
  - Viewing categorized inbox
  - Reading summaries
  - Editing AI-generated replies

---

6. Workflow

1. User connects email account
2. System retrieves incoming emails
3. Emails are preprocessed
4. AI modules perform:
    - Summarization
    - Classification
    - Priority scoring
    - Reply generation
5. Results displayed in dashboard
6. User reviews or edits responses
7. Email is sent (manual or automated)

---

7. Technology Stack

Backend

- Python / Node.js
- FastAPI / Express

AI & NLP

- Transformer-based models (e.g., GPT)
- Hugging Face ecosystem
- spaCy / NLTK

Frontend

- React / Next.js
- Tailwind CSS

Database

- PostgreSQL / MongoDB

Integrations

- Gmail API
- Microsoft Graph API

---

8. Security & Privacy

- OAuth2-based authentication
- Encrypted data transmission (HTTPS)
- Optional minimal data retention policy
- Compliance-ready design (GDPR considerations)

---

9. Evaluation Metrics

- Classification Accuracy
- Summary Quality (ROUGE / BLEU scores)
- Response Relevance
- User Satisfaction Score

- Time saved per email

---

10. Challenges & Mitigation

Challenge	Mitigation
Long email threads	Context window optimization
AI hallucination	Prompt constraints + validation
Misclassification	Continuous model fine-tuning
User trust	Editable AI outputs

---

11. Future Enhancements

- Voice-based summaries
- Calendar and task integration
- Team collaboration features
- Personalized learning models
- Sentiment analysis
- Auto-follow-up reminders

---

12. Use Cases

Professionals

- Manage high email volumes efficiently

Customer Support Teams

- Automate ticket classification and responses

Managers & Executives

- Quickly review summarized communication threads

---

13. Impact

- Significant reduction in email processing time
- Improved response efficiency
- Enhanced decision-making through summarized insights
- Reduced cognitive load

---

14. Conclusion

The Email Triage Assistant demonstrates how AI can be effectively applied to real-world productivity challenges. By combining NLP, automation, and intelligent design, this system transforms email management into a streamlined, efficient process.

It showcases strong competencies in:

- AI system design
- NLP implementation
- Full-stack development
- Product thinking