# AI-Powered Email Classification & Metadata Extraction

## Problem Statement

In commercial banking, loan servicing teams receive thousands of email requests daily. These emails often contain attachments and require manual triage by a gatekeeping team that classifies them, extracts key data, and assigns them to the right teams. This process is:  
- ❌ Time-consuming  
- ❌ Error-prone  
- ❌ Operationally expensive  
  
To automate this, we built an AI-powered solution that classifies emails, extracts metadata from both email body and attachments, and detects duplicate emails using similarity search with embeddings.

## Inspiration

- Manual email processing in banking is inefficient, leading to high operational costs and errors.  
- Generative AI models (LLMs) can understand unstructured text, making them ideal for automating classification.  
- OCR & Embeddings enable us to process PDFs, images, and detect duplicate emails effectively.  
- AI-driven automation in financial services inspired us to build a scalable, explainable, and efficient solution.

## Data Source

- \*\*Emails:\*\* `.eml` files stored in a directory.  
- \*\*Attachments:\*\* PDFs, DOCX, images (JPEG, PNG), and nested `.eml` files.  
- \*\*Classification Data:\*\* Predefined request types and sub-request types from banking workflows.  
- \*\*Vector Storage:\*\* ChromaDB for similarity search on embeddings.

## Solution

### Features:  
✔ Email Ingestion & Parsing: Extracts email content and attachments, handling nested emails recursively.  
✔ AI-Powered Classification: Uses GPT-4-turbo to classify emails into predefined categories.  
✔ Metadata Extraction: Extracts key fields dynamically based on request type & sub-request type.  
✔ Duplicate Detection: Uses sentence-transformers and ChromaDB to find similar emails.  
✔ Interactive Gradio UI: Displays classification results and metadata with real-time progress tracking.

## Design & Architecture

### Architecture Diagram:  
\*(Placeholder: Add architecture diagram here)\*  
  
\*\*Workflow:\*\*  
1️⃣ Email Ingestion: Parses `.eml` files, extracts content, and identifies attachments.  
2️⃣ AI Classification: GPT-4-turbo assigns request type & sub-request type.  
3️⃣ Metadata Extraction: Key fields are extracted from email body & attachments.  
4️⃣ Duplicate Detection: Embeddings stored in ChromaDB help find duplicate emails.  
5️⃣ Results Displayed: Gradio UI presents classification results in real time.

## Screenshots

\*(Placeholder: Add UI screenshots here)\*

## Challenges We Faced

🔴 Handling Nested Emails: Gmail `.eml` files required a fix for `message/rfc822` parsing.  
🔴 Extracting Key Metadata Dynamically: Fields vary by request type, so we split classification & metadata extraction.  
🔴 OCR Accuracy on Images: Switched from `pytesseract` to `easyocr` for improved results.  
🔴 Duplicate Detection Tuning: Optimized similarity threshold to 0.90 to reduce false positives.

## Tech Stack Used

✅ \*\*Frontend & UI:\*\* Gradio  
✅ \*\*AI & NLP:\*\* GPT-4-turbo, Function Calling API  
✅ \*\*Document Processing:\*\* pdfplumber, python-docx, easyocr  
✅ \*\*Vector Search & Duplicate Detection:\*\* sentence-transformers, ChromaDB  
✅ \*\*Backend & Parsing:\*\* email, BeautifulSoup

## Contributors & Contact

📧 Email: [Your Email]   
🔗 GitHub: [Your GitHub Link]