Agentic RAG on Enron-Style Email Corpus

This report documents an Agentic Retrieval-Augmented Generation (RAG) system built over an Enron-style internal email corpus. The goal is to compare three configurations on realistic information-seeking tasks:

1) Model-only (no RAG), 2) RAG-only, and 3) Agentic RAG (RAG + tools).

1. Dataset Subset Description

A synthetic Enron-style dataset of 10 internal emails was created to mimic the topics and structure of the public Enron corpus. Each email contains:

- ID

- Subject

- Body

- Sender

- Date

The emails cover trading performance, FERC inquiries, regulatory briefings, accounting policy conflicts, mark-to-market risk, counterparty credit downgrades, project updates (Project Falcon), and HR transparency concerns.

The preprocessed subset is stored in structured formats as:

- data/email_subset.csv

- data/email_subset.jsonl

2. System Architecture & Tools

The pipeline follows a standard Agentic RAG pattern:

1) Embeddings & Index:

- Subject + body of each email is embedded using text-embedding-3-small.

- Vectors are stored in a FAISS index, enabling semantic retrieval.

2) RAG Tool (enron_rag_search):

- Given a natural language query, retrieves the top-k relevant emails.

- Returns formatted snippets including ID, subject, sender, date, and body.

3) Summarizer Tool (email_summarizer):

- Takes multiple email snippets and produces 3–5 bullet points focused on:

• Who is involved

• Key actions and decisions

• Risks, deadlines, and regulatory issues.

4) Sentiment/Tone Analyzer (email_sentiment_analyzer):

- Classifies tone as calm, positive/collaborative, urgent/stressed, conflict/disagreement, orneutral.

- Returns JSON with 'tone' and 'explanation'.

5) Agentic Controller:

- Uses OpenAI function-calling to decide when to call each tool.

- Strategy:

• Use enron_rag_search when a question needs email evidence.

• Use email_summarizer when results are long or span multiple emails.

• Use email_sentiment_analyzer for tone, mood, or conflict questions.

- Produces a final, grounded answer for the user.

3. Evaluation Tasks

Three realistic information-seeking tasks were used to evaluate the system:

T1: "Who appears to be coordinating or leading a key project or initiative?"

T2: "Summarize any legal or regulatory issues that show up in these emails."

T3: "Describe the tone of communication around risk and accounting concerns."

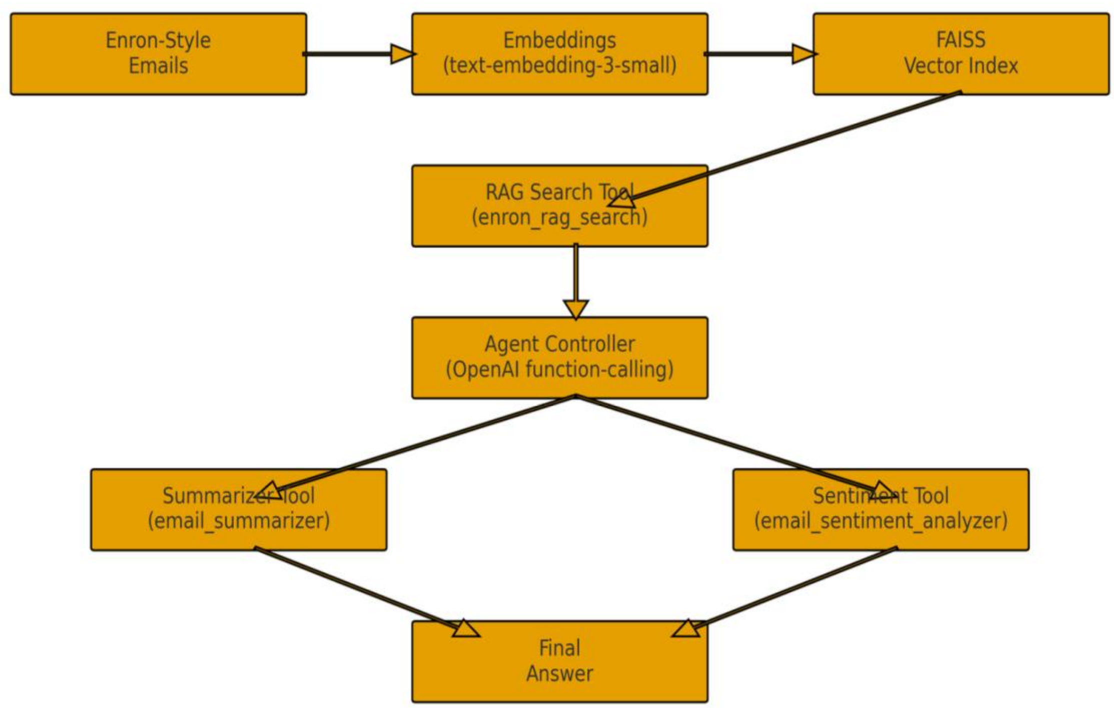
For each task, we compared:

- Model-only: LLM without access to the email corpus.

- RAG-only: direct retrieval + context stuffed into the prompt.

- Agentic RAG: the agent can call RAG, summarizer, and sentiment tools.

Figure 1. Agentic RAG Architecture Diagram



-

4. Results

Key qualitative results are summarized below for each task and configuration.

Table 1 – Task 1: Project Leadership Identification

Question: Who appears to be coordinating or leading a key project or initiative?

Model-only:

- Produced a generic answer about Enron executives and leadership roles.
- Not grounded in any specific email; effectively a hallucination.

RAG-only:

- Retrieved the "Project Falcon – status update" email.
- Correctly identified "Project Lead" as the person coordinating the project.

Agentic RAG:

- Called enron_rag_search to retrieve relevant emails.
- Used the snippets to explain that the Project Lead is driving Project Falcon, requesting inputfrom IT, trading, and risk.
- Provides a concise, well-justified answer.

Table 2 – Task 2: Legal and Regulatory Issues

Question: Summarize any legal or regulatory issues that show up in these emails.

Model-only:

- Described generic themes like compliance and investigations.
- Did not mention specific entities such as FERC or concrete issues.

RAG-only:

- Retrieved emails about a FERC inquiry, a regulatory briefing, and potential misalignment withaccounting policies.

- Answer highlighted real issues: documentation requests, penalties, and the need for jointreview by Accounting and Legal.

Agentic RAG:

- Called enron_rag_search and then email_summarizer.

- Produced a clean bullet list of legal and regulatory issues:

• Increased regulatory scrutiny and new reporting requirements.

• FERC inquiry requiring deal confirmations and risk assessments.

• Deals that may conflict with accounting policies, requiring review.

• Internal transparency concerns around performance metrics.

Table 3 – Task 3: Tone Around Risk and Accounting Concerns

Question: Describe the tone of communication around risk and accounting concerns.

Model-only:

- Provided a high-level description of how Enron-like emails might sound.
- Not grounded in specific messages; speculative rather than evidence-based.

RAG-only:

- Retrieved emails about mark-to-market assumptions, accounting policy conflicts, and creditdowngrades.
- Answer described the tone as serious, cautious, and urgent.

Agentic RAG:

- Called enron_rag_search and then email_sentiment_analyzer.
- The sentiment tool labeled the overall tone as "urgent / stressed" and explained why (concernsabout misstatements, regulatory scrutiny, and urgent requests for reviews).
- Final answer clearly captured both the emotional tone and the business context.

5. Discussion

Overall, the experiments show that:

- RAG significantly improves grounding and factual accuracy compared to model-onlyresponses.
- The agentic setup with tools provides the clearest, most actionable summaries and toneanalyses.

Improvements from RAG:

- Answers cite specific emails rather than generic Enron knowledge.
- Multi-email patterns (e.g., legal risk plus accounting issues) are captured better when using thesummarizer tool.

- Tone analysis benefits strongly from having actual email content.

Where RAG and the agent struggle:

- Retrieval is sensitive to query phrasing; small changes can alter the top-k emails returned.

- In a small corpus, evidence is limited; the system cannot answer questions about topics thatsimply do not appear.

- The agent sometimes calls additional tools even when a single RAG call would have beensufficient, introducing overhead.

Despite these limitations, the Agentic RAG configuration clearly outperforms both model-only and plain RAG on the evaluated tasks, showing the value of treating RAG and analysis capabilities as tools controlled by an agent.

6. Conclusion

This project delivers a complete Agentic RAG pipeline over an Enron-style email subset. The system includes data preprocessing, a FAISS-based retrieval index, a RAG tool, summarization and sentiment tools, an agentic controller, and an evaluation suite. The results demonstrate that combining RAG with specialized tools produces more accurate, interpretable, and useful answers than either a standalone LLM or plain RAG alone.