

Email Summarizer API Documentation

July 18, 2025

Contents

1	Introduction	1
2	Sample Input and Output	1
2.1	Sample Input 1	1
2.2	Sample Output 1	2
2.3	Sample Input 2	2
2.4	Sample Output 2	3
3	Code Structure	3
4	Model Description	4
5	Flow	4
6	API Endpoint	4
6.1	Request Body: EmailThread Schema	4

1 INTRODUCTION

This is a FastAPI-based application that summarizes email threads using Google Gemini (via LangChain). It intelligently processes and condenses multiple emails into concise summaries while preserving key information like sender names, subject, and intent.

2 SAMPLE INPUT AND OUTPUT

2.1 SAMPLE INPUT 1

```
{
  "thread_id": "thread-123",
  "email_count": 2,
  "emails": [
    {
      "id_": "email-1",
```

```
{
  "from_": "alice@example.com",
  "to": ["bob@example.com"],
  "cc": [],
  "bcc": [],
  "subject": "Project Update",
  "body": "Hey Bob, here's the latest on the project...",
  "is_parent": true,
  "parent_email_id": ""
},
{
  "id_": "email-2",
  "from_": "bob@example.com",
  "to": ["alice@example.com"],
  "cc": [],
  "bcc": [],
  "subject": "Re: Project Update",
  "body": "Thanks Alice. Looks good. I added a few suggestions...",
  "is_parent": false,
  "parent_email_id": "email-1"
}
]
```

2.2 SAMPLE OUTPUT 1

```
{
  "summaries": {
    "thread-123": [
      "Alice shared the latest updates on the project.",
      "Bob appreciated the updates and added suggestions."
    ]
  }
}
```

2.3 SAMPLE INPUT 2

```
{
  "thread_id": "thread-124",
  "email_count": 2,
  "emails": [
    {
      "id_": "email-1",
      "from_": "alice@example.com",
      "to": ["bob@example.com"],
      "cc": [],
      "bcc": [],
      "subject": "Project Update",
      "body": "Hey Bob, here's the latest on the project...",
      "is_parent": true,
      "parent_email_id": ""
    },
    {
      "id_": "email-2",
```

```
{
  "from_": "bob@example.com",
  "to": ["alice@example.com"],
  "cc": [],
  "bcc": [],
  "subject": "Re: Project Update",
  "body": "Thanks Alice. Looks good. I added a few suggestions...",
  "is_parent": false,
  "parent_email_id": "email-1"
}
```

2.4 SAMPLE OUTPUT 2

Note: The Sample Output 2 below is reproduced as provided, but it contains inconsistencies: it references "thread-123" (should be "thread-124") and mentions "Carol" and "Dave" (not present in the input). A corrected version would align with the input for "thread-124" using "Alice" and "Bob."

```
{
  "summaries": {
    "thread-123": [
      "Alice shared the latest updates on the project.",
      "Bob appreciated the updates and added suggestions."
    ],
    "thread-124": [
      "Carol sent the meeting agenda for tomorrow's meeting and asked if Dave
        wanted to add anything.",
      "Dave thanked Carol for the meeting agenda and suggested adding a
        discussion about last week's client feedback."
    ]
  }
}
```

3 CODE STRUCTURE

The project is organized as follows:

- email-summarizer-api/
 - app/
 - `__init__.py`: Package initializer.
 - `email_summary_FastAPI.py`: Core logic: prompt setup and Gemini integration.
 - `.env`: Environment variables (e.g., `GOOGLE_API_KEY`).
 - `requirements.txt`: Python dependencies.
 - `README.md`: Project documentation.

4 MODEL DESCRIPTION

This application uses:

- **Model:** gemini-2.5-flash via LangChain's ChatGoogleGenerativeAI.
- **Purpose:** To summarize email bodies and subjects in 1–2 lines.
- **Behavior:**
 - For the parent email, a direct summary is generated.
 - For follow-up emails, the model is given previous summaries for context.
 - The summaries are stored thread-wise using an in-memory dictionary.

5 FLOW

The application processes email threads as follows:

- User sends a POST request to /email/summarize with a list of emails in a thread.
- The app processes each email:
 - If it's a parent email, it generates a summary directly.
 - If it's a child, it prepends earlier summaries for contextual summarization.
- The response contains a list of thread-wise summaries. For example:
 - email1 → s1
 - email2 → s1 + email2(body) → s2
 - email3 → s1 + s2 + email3(body) → s3

6 API ENDPOINT

POST /email/summarize

6.1 REQUEST BODY: EMAILTHREAD SCHEMA

```
{
  "emails": [
    {
      "id_": "string",
      "from_": "string",
      "to": ["string"],
      "cc": ["string"],
      "bcc": ["string"],
      "subject": "string",
      "body": "string",
      "is_parent": true/false,
      "parent_email_id": "string"
    }
  ],
}
```

```
"thread_id": "string",  
"email_count": int  
}
```