

AI-Powered Personal Email Assistant

Introduction

This project is a sophisticated **local AI-powered automation ecosystem** designed to streamline your daily digital intake while maintaining total data privacy. By integrating a personal Gmail account with a **locally hosted Large Language Model (Ollama + LLaMA)**, the system autonomously fetches, summarizes, and categorizes incoming correspondence into dedicated streams such as Jobs, Finance, and Newsletters.

Each morning, the system compiles these insights into a refined **Telegram briefing**, enriched with real-time weather reports and the latest AI industry news. Because the intelligence layer runs entirely on your own hardware, the system ensures that sensitive communications never leave your local environment, offering a seamless blend of high-level productivity and **cloud-free security**.

Project Description

This project is a **local AI-powered email automation system** that:

- Securely reads emails from a personal Gmail account
- Uses a **locally running LLM (Ollama + LLaMA)** to:
 - Summarize each email intelligently
 - Categorize emails into Jobs, Bank, Newsletters, and Miscellaneous
- Generates a **daily morning briefing**
- Sends the formatted summary to **Telegram**
- Adds **Top AI News** and **Weather Report**
- Runs completely **locally** with **no cloud AI usage**

⚠️ **No email content is sent to external AI services**

⚠️ **No paid APIs used**

Project Objectives

- Reduce time spent reading emails
 - Highlight important emails and required actions
 - Maintain complete data privacy
 - Use only **free and open-source tools**
 - Build a production-like AI automation system
-

Tools & Technologies Used

Programming Language

- **Python 3.11.x** (Recommended & stable)

AI / LLM

- **Ollama**
- ****LLaMA 3**

APIs & Libraries

- Gmail API
- Telegram Bot API
- wttr.in (free weather)
- RSS feeds (AI news)

Python Libraries

- google-api-python-client
- google-auth
- google-auth-oauthlib
- python-telegram-bot
- requests
- feedparser

Platform

- Windows Laptop (Local Execution)
 - VS Code
-

Core Code Components

◆ **gmail_reader.py**

- Handles OAuth login
 - Saves token locally
 - Prevents repeated logins
 - Fetches recent email snippets
-

◆ **email_summarizer.py**

- Uses Ollama
 - Summarizes email content
 - Classifies into categories
 - Uses example-driven prompting for accuracy
-

◆ **summarizer.py**

- Groups emails by category
 - Limits number per section
 - Formats final Telegram message
-

◆ telegram_bot.py

- Handles async Telegram messaging
 - Splits long messages safely
 - Prevents timeout errors
-

◆ ai_news.py

- Fetches latest AI headlines using RSS
 - No API keys required
-

◆ weather.py

- Fetches real-time weather
 - Uses free public service
-

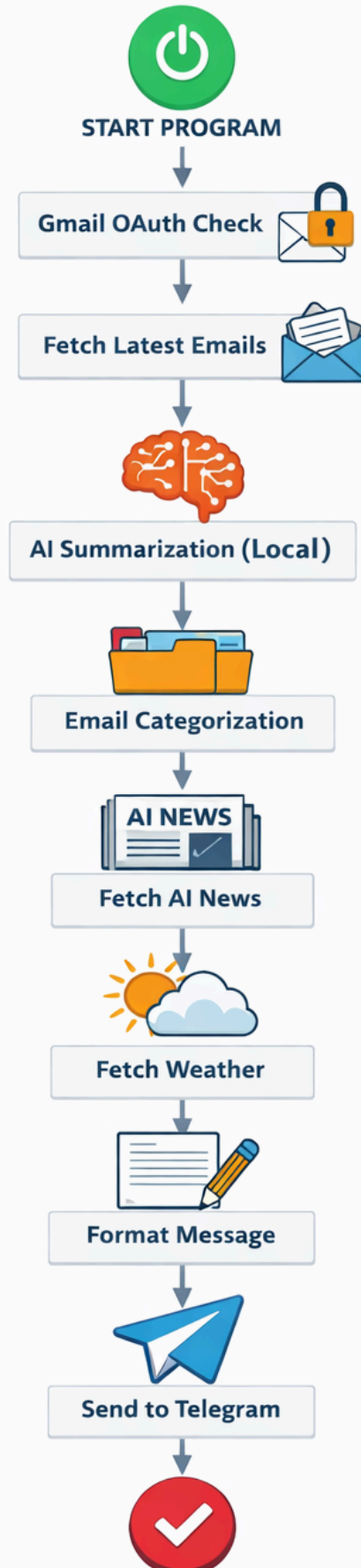
◆ main.py

- Orchestrates the entire pipeline
 - Entry point for execution
-

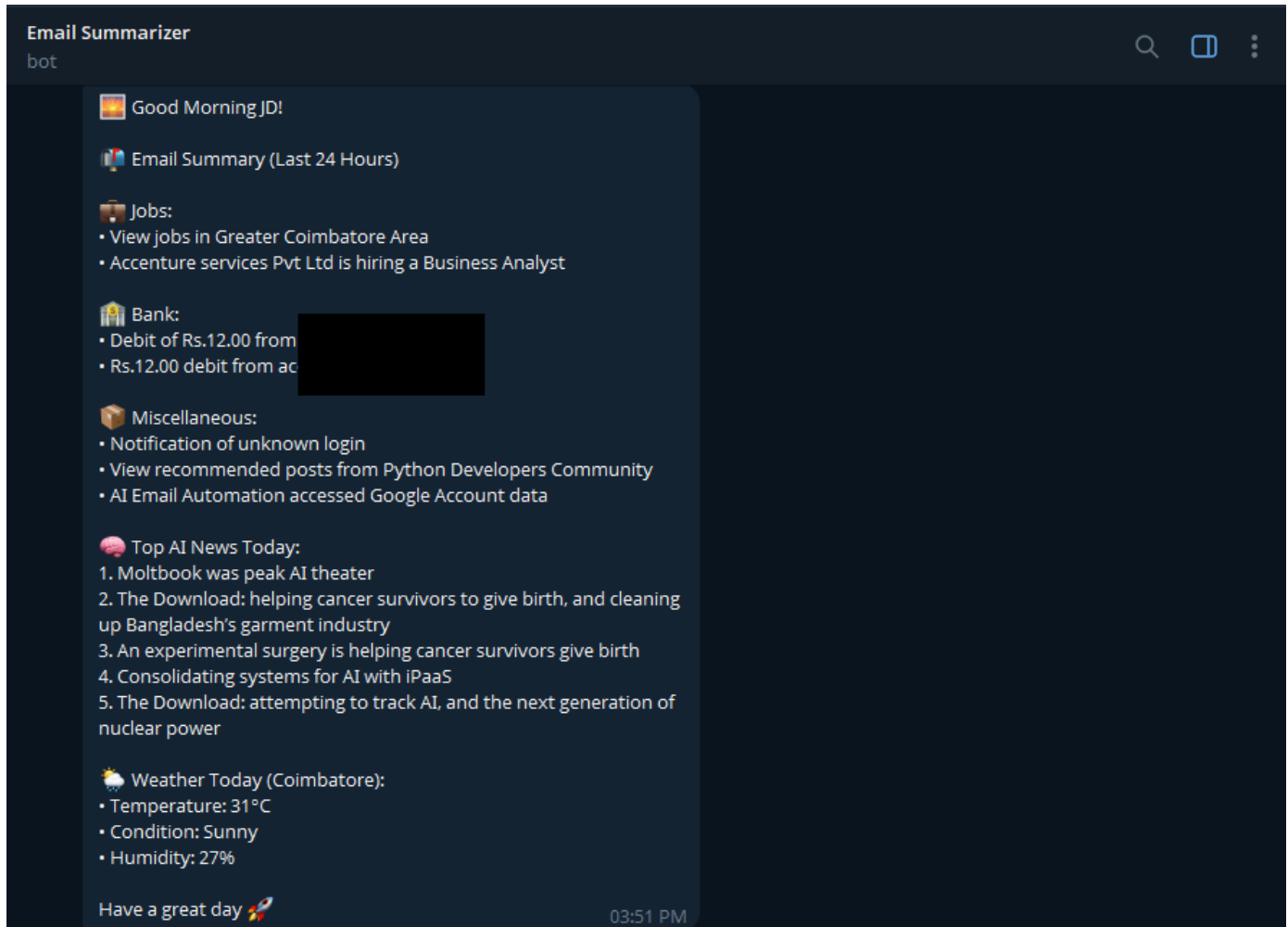
Execution Flow :



Automated Email Summary Process



Output



Conclusion

Instead of sending your personal data to a big tech company's cloud, everything happens right on your own computer. You get the power of modern AI to sort your mail, check the weather, and catch you up on the news, all while knowing your private information stays exactly where it belongs with you. It's a smarter, faster, and much more secure way to handle the daily "email grind."