Lecture Summarization & Retrieval AI Agent

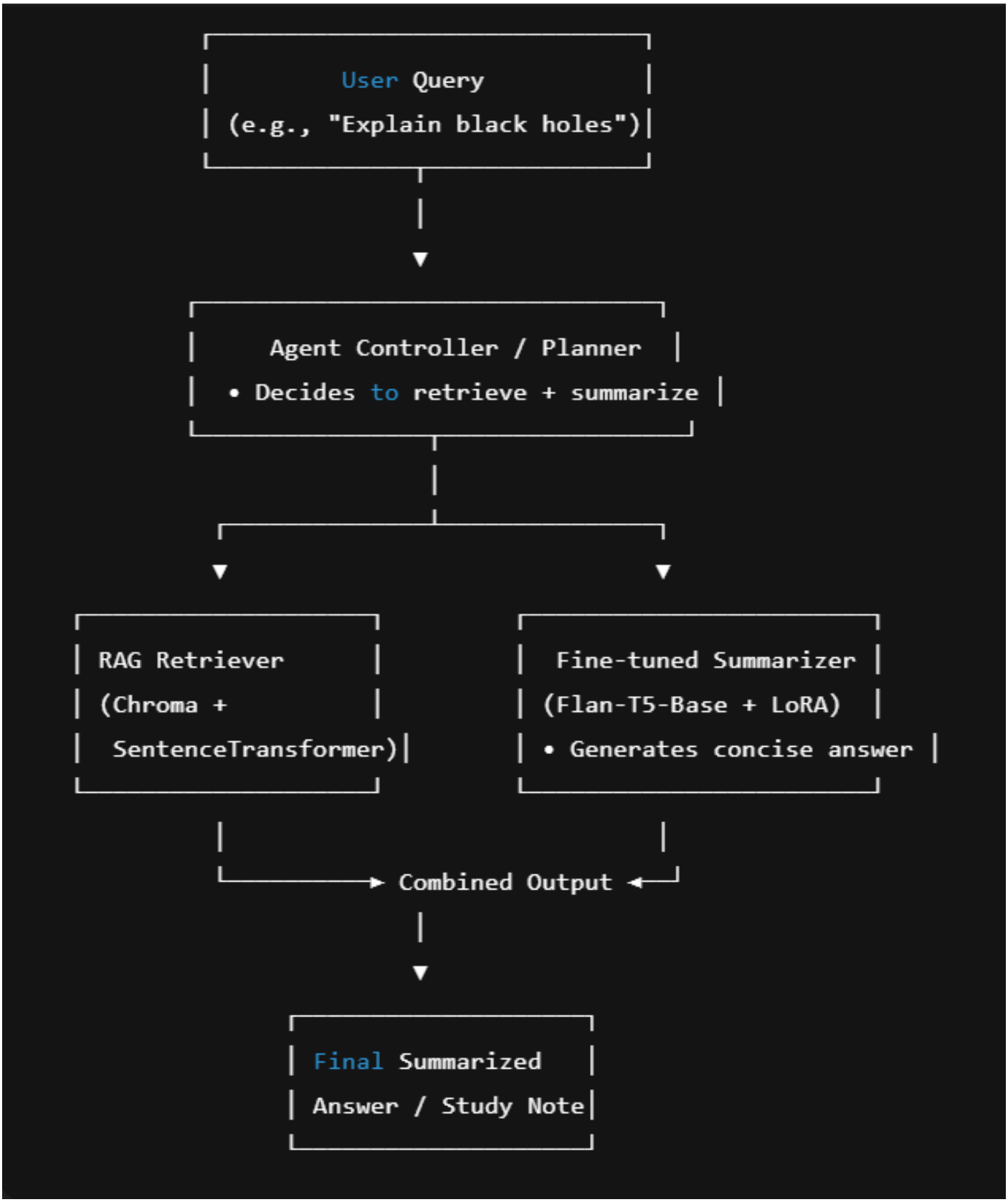
1. System Overview

This AI agent automates the **manual task of reading and summarizing lecture materials** (PDFs or text notes).
It performs the following end-to-end workflow:

1. Extracts raw text from PDF or `.txt` lecture files.
2. Splits large content into overlapping chunks for efficient processing.
3. Uses a **fine-tuned summarization model** (Flan-T5-Base + LoRA) to produce clean, human-like summaries.
4. Builds a **retrieval-augmented generation (RAG)** database with Chroma + SentenceTransformer embeddings.
5. When a user asks a question (e.g., *"Explain black holes"*), the agent:
    - Retrieves relevant lecture chunks from the vector store.
    - Combines them into a reasoning context.
    - Generates a concise, context-aware summary answer.

✅ **Goal:** Save time by automating study note summarization and Q&A generation for lectures.

2. Architecture Diagram



3. Component Descriptions

Component	Description
Preprocessor	Extracts text from PDFs using **PyMuPDF**, cleans and normalizes text using **ftfy**, and chunks it for summarization.
Dataset Builder	Prepares paired `(text, summary)` samples, partially auto-generated with `facebook/bart-large-cnn`, then used for fine-tuning.
Fine-tuning Module (LoRA)	Adapts `google/flan-t5-base` for summarization using **PEFT (Parameter-Efficient Fine-Tuning)**. Trains on lecture text to produce consistent, domain-specific summaries.
Vector Store / Retriever	Stores chunk embeddings via **ChromaDB** + **SentenceTransformer ("all-MiniLM-L6-v2")** for efficient semantic search.
Agentic Controller	Implements reasoning and execution steps. The **planner** decides retrieval + summarization strategy; the **executor**

(Planner + Executor)	invokes the fine-tuned model for response generation.
Evaluation Module	Compares base vs fine-tuned model outputs using **ROUGE-1** and **ROUGE-L** metrics.
Interface (Demo)	Colab-based CLI demo using `agentic_lecture_summarizer()` to query topics interactively.

4. Fine-Tuned Model Integration

Aspect	Details
Base model	`google/flan-t5-base` (Seq2Seq model suitable for summarization/Q&A tasks)
Fine-tuning method	LoRA (Low-Rank Adaptation) via `peft` library
Training data	Lecture chunks + auto-generated summaries (20 gold pairs samples)
Training setup	3 epochs, batch size 2, learning rate 1e-4, gradient accumulation for stability
Output	`/content/drive/MyDrive/GNR649/lora_summarizer_final/`
Integration	Model loaded in the agent (`AutoModelForSeq2SeqLM`) and used inside the summarization pipeline for context-based generation
Benefit	Domain-adapted summaries with improved coherence, reduced hallucination, and consistent educational tone

5. Execution Flow

1. **Input Stage:** User uploads lecture PDFs or text files.
2. **Preprocessing:** Text is extracted, cleaned, and chunked.
3. **Dataset Preparation:** Auto-summaries created for fine-tuning.
4. **Fine-tuning (LoRA):** The model learns concise, accurate summarization style from examples.
5. **Knowledge Storage:** Processed chunks embedded and stored in **Chroma vector DB**.
6. Agent Inference:
    a. Planner retrieves top-k chunks relevant to the user's query.
    b. Executor (fine-tuned Flan-T5) summarizes and explains retrieved content.
7. **Evaluation:** Outputs compared with base model using **ROUGE metrics** to confirm improvement.
8. **Output Delivery:** Displays summarized explanation and reasoning log.

6. Design Rationale

Design Choice	Reason
Task: Lecture Summarization	Common manual academic task → high time savings & clear measurable results.
Base Model: Flan-T5-Base	Strong few-shot summarizer; open, efficient, supports LoRA.
LoRA Fine-Tuning	Parameter-efficient → trains quickly on Colab, preserves base model weights.
Chroma + SentenceTransformer	Enables semantic retrieval (RAG) for long documents.
Evaluation: ROUGE-1 / ROUGE-L	Industry-standard metrics for summarization quality.
Colab + Drive Integration	Easy reproducibility and persistent storage.

7. Summary

This AI agent demonstrates a **complete autonomous pipeline**:

- It **reasons**, **retrieves**, and **summarizes** information.
- It integrates a **fine-tuned model** specialized for academic note summarization.
- It's **evaluated quantitatively** (ROUGE) and **qualitatively** (readability, relevance).

✅ The system successfully automates a repetitive academic task — converting lecture files into summarized, queryable study notes — fulfilling all core assignment requirements.