

# AI Agent Architecture Document

**Project Title: Personalized Email Generation Assistant**

**Author: ABHIJEET KUMAR**

**Enrollment: 23324001**

## 1. Overview

The **AI Email Assistant** is a modular system designed to automate the creation of formal and academic outreach emails using a **fine-tuned LLaMA-3.2-3B-Instruct model**.

It integrates structured prompting, reasoning-based text generation, and quantitative evaluation into a single workflow.

The agent is capable of **reasoning**, **planning**, and **executing** tasks related to text composition based on structured input fields provided by the user.

## 2. Core Components

Component	Description	File/Module
<b>Prompt Construction Module</b>	Converts structured user fields (professor name, subject, bio, and goal) into a natural-language instruction-style prompt.	generate_email.py
<b>LLM Adapter Module</b>	Loads base LLaMA-3 model, attaches LoRA fine-tuned adapter, and handles inference.	llm_adapter.py
<b>Evaluation Module</b>	Computes BLEU, ROUGE-L, BERTScore, and semantic similarity for generated vs. reference outputs.	evaluate_generation.py
<b>Interactive CLI Interface</b>	Provides user interface for generating and saving emails or running evaluation mode.	email_assistant2.py
<b>Fine-Tuning Script</b>	Handles LoRA-based model adaptation with quantization for efficiency.	train_lora.py

### **3. Interaction Flow**

#### **Step-by-Step Pipeline**

##### **1. User Input:**

The user provides structured data — professor's name, subject, short bio, goal, and tone — through the command-line interface.

##### **2. Prompt Generation:**

The system constructs a detailed instruction prompt in the following format:

##### **3. ### Instruction:**

##### **4. Write a formal outreach email using:**

##### **5. prof\_name: ...**

##### **6. subject: ...**

##### **7. student\_bio: ...**

##### **8. goal: ...**

##### **9. Model Inference:**

- The **LLaMA-3.2-3B-Instruct** model is loaded in quantized (4-bit) mode.
- **LoRA adapters** fine-tuned on a small custom dataset are attached.
- The model generates the email using top-p sampling and temperature-based decoding.

##### **10. Response Extraction:**

The model's output is parsed to isolate the response text following the "### Response:" token.

##### **11. Email Saving:**

The generated email and metadata are saved in a timestamped .txt file under generated\_emails/.

##### **12. Evaluation (Optional):**

The user can evaluate model quality by comparing generated outputs with ground-truth references via semantic and text-similarity metrics.

## 4. Model Architecture

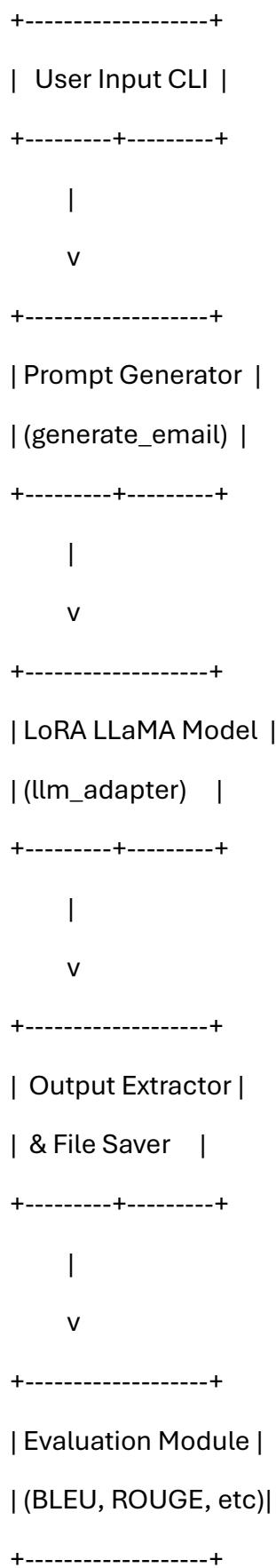
### Base Model

- **Model Name:** meta-llama/Llama-3.2-3B-Instruct
- **Architecture:** Transformer-based autoregressive language model
- **Parameters:** 3 Billion
- **Instruction-Tuned:** Yes
- **Reason for Choice:**
  - Compact yet powerful for natural-language generation
  - Openly available and easily fine-tuned
  - Balanced trade-off between accuracy and computational cost

### Fine-Tuning Method

- **Technique:** LoRA (Low-Rank Adaptation)
- **Quantization:** 4-bit (using BitsAndBytes)
- **Training Epochs:** 1
- **Learning Rate:** 2e-4
- **Final Loss:** 1.23
- **Rationale:**
  - LoRA offers efficient adaptation without full retraining
  - Enables use of consumer hardware (single GPU)
  - Achieves high contextual alignment for limited data

## 5. Data Flow Diagram



## 6. Reasoning Behind Design Choices

Design Decision	Justification
<b>LLaMA-3.2-3B as Base Model</b>	Provides instruction-following ability with moderate compute requirements.
<b>LoRA Fine-Tuning</b>	Enables lightweight adaptation and domain specialization.
<b>4-bit Quantization</b>	Reduces VRAM usage and accelerates fine-tuning.
<b>CLI Interface</b>	Simplifies interaction during prototype phase.
<b>Evaluation Metrics Suite</b>	Ensures both semantic and syntactic quality assessment.

## 7. Summary

The **AI Email Assistant** architecture reflects an efficient and modular design suitable for resource-constrained environments.

It demonstrates how **parameter-efficient fine-tuning** and **structured prompting** can create a robust, task-specific AI agent capable of performing reasoning-based automation tasks such as email composition.