UMAKSHI  
AI Scientist, Data Scientist, ML Engineer,Python Developer

Kururkshetra, Haryana

***ABOUT\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

(An AI enthusiast, leverage in artificial intelligence and looking for the opportunities to work in LLM training, Fine-Tuning and Langchain flow. To use cutting edge technologies in companies solution to solve real life problems.)

Aspiring AI & Data Science professional with a strong interest in LLM training, AI agent development, and data-driven solutions. Familiar with machine learning, data analysis, and model building using Python, Pandas, NumPy, and Scikit-Learn. Eager to learn and apply AI techniques to real-world problems, improve ML models, and develop scalable AI applications. Looking for opportunities to grow and contribute to innovative AI projects.

***PROFESSIONAL EXPEREINCE\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

***AI Scientist  
( iAssist Innovations labs) ( April’24 - May’25)***

· Spearheaded the enhancement of mICRa, a collaborative product with ICICI Lombard, by automating monitoring processes and integrating new features. (Skills: Product Enhancement, Process Automation | Technologies: Python, Flask, ICR/OCR, Pandas)

· Took charge of the reporting module, demonstrating proficiency in Pandas to generate a variety of reports, showcasing analytical capabilities. (Skills: Data Analytics, Reporting | Technologies: Pandas, Python)

· Converted a monolithic application into micro-services and dockerized the application, significantly improving scalability and maintainability. (Skills: Microservices, Docker | Technologies: Docker, Microservices Architecture)

· Worked on improving prediction accuracy by refining algorithm parameters and Data. (Skills: Machine Learning, Model Optimization | Technologies: Python, Data Preprocessing)

· Worked with different clients of MIBL, ICICI Lomabard and build the FRS for their product by understand their needs. (Skills: Client management, Product development, Project management)

**ML/AI INTERN (CSIR - CSIO\_)**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Developed an ”AI CAR ASSISTANT” project that predicts vehicle acceleration based on dataset features.
2. Utilized data mining techniques for analyzing, visualizing, and preprocessing real-time series datasets by using correlation and interpolation techniques.
3. Implemented Multiple Linear Regression algorithm and evaluated results using RMSE, MSE, and R2 Scores.
4. Trained models using the Cross-validation technique, achieving an impressive R2 score of 0.9574.

***ANDROID DEVELOPER (OOHR INNOVATIONS\_)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

· Accomplished Android Developer Intern at OOHR INNOVATIONS. (Skills: Android Development, Mobile App Development | Technologies: Kotlin, XML, Android Studio)

· · Contributed to the creation of a driver’s app by preparing workflow and flowcharts, showcasing project collaboration and effective app design strategies. (Skills: UI/UX Design, Project Collaboration | Technologies: Figma, Wireframing, Flowcharting)

***DATA SCIENTIST INTERN (EXPOSYS DATA LABS\_)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

· Developed a predictive model for company profits using machine learning to analyze R&D, Administration, and Marketing Spend. (Skills: Machine Learning, Predictive Analytics | Technologies: Linear Regression, Decision Trees, Neural Networks)

· Implemented multiple regression algorithms, including Linear Regression, Lasso, Ridge, Decision Tree, Random Forest, and Support Vector Regression (SVR), for performance optimization. (Skills: Regression Analysis, Model Evaluation | Technologies: Scikit-Learn, Python)

· Conducted exploratory data analysis (EDA) to preprocess and visualize company financial data for improved model performance. (Skills: Data Analysis, Feature Engineering | Technologies: Pandas, NumPy, Seaborn, Matplotlib)

· Optimized model accuracy, achieving 97.95% using Linear Regression, evaluated through MSE, MAE, and R² score. (Skills: Model Optimization, Performance Tuning | Technologies: Python, Scikit-Learn)

· Deployed the machine learning model using Streamlit, creating an interactive web interface for real-time profit prediction. (Skills: Model Deployment, Web Development | Technologies: Streamlit, Flask, Python)

· Automated model training and deployment, ensuring scalability and efficiency in predictive analytics. (Skills: Automation, ML Pipelines | Technologies: Python, Scikit-Learn, Pickle)

SHADOVEIN

· Interned as a Subject Matter Expert (Mathematics) at Shadovein, applying mathematical expertise to solve real-world problems and provide insights. (Skills: Problem-Solving, Analytical Thinking | Technologies: Mathematics, Quantitative Analysis)

· · Developed proficiency in MS Office and Content Management Systems (CMS), gaining practical experience in job ethics and professional communication. (Skills: Documentation, Content Management | Technologies: MS Office, CMS)

SKILLS\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Programming Languages: Python(proficient),C/C++

Libraries and Frameworks: Pandas, Scikit Learn,Flask, Matplotlib, Streamlit, Tensorflow, Keras, CNN, Docker, CI/CD using Jenkins, Heroku.

Developer Tools: Git, Github, Postman, Visual Studio Code (IDE).

Database: MySQL

Prompt Engineering: GPT-4o,Dall-e,MidJourney,Pixlr,DeepSeek r1

EDUCATION

UIET, Kurukshetra University Aug 2020 – June 2024

1. Tech - Electronics and Communication - CGPA - 8.54 Kurukshetra, India

***ACADEMIC PROJECTS***

DRY HERBS - Ayurvedic herbs classifier March 2024 - Present

• Gained hands-on experience in a project with PhD scholar from Shri Krishna Ayush university, focusing on the use of computer vision technology to classify Ayurvedic herbs, stems, fruits, and seeds.

• Learned the full project development process, including dataset collection, preprocessing, and the application of data augmentation techniques to expand the dataset.

• Acquired knowledge in using Convolutional Neural Network (CNN) models for image classification and in developing APIs with Django.

• Understood the integration of a Large Language Model (LLM) API to enhance the accuracy of the classification system.

POSITION OF RESPONSIBILITY

CONTENT WRITER LEAD at E-Cell UIET KUK Kurukshetra, India

May 2022 - May 2024

• Accomplished Content Writer Lead at E-Cell UIET KUK, orchestrating events and activities impacting over 1000 students, promoting Innovation and Entrepreneurship in the Institute, measured by organizational skills and event management capabilities.

Technical Expertise  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

I am proficient in Python and have also worked with C.

More specifically in Python for Data pre-processing i worked with Pandas, Numpy and Polars for fast Processing. For Data visualization i used Matplotlib and Seaborn.

In system architecture, I have experience with both Monolithic and Microservices architectures and have Dockerized applications.

For databases, I have worked with MySQL.

In Machine Learning projects, I have primarily worked with Regression algorithms using scikit-learn (sklearn). For model deployment, I have used Pickle.

For Python application UIs, I have used Streamlit, while for my portfolio website, I developed it using HTML, CSS, and JavaScript.

For server automation, I have used Cron jobs and Service files.

In CI/CD deployment, I have worked with Jenkins and GitHub for applications, while for websites, I used Netlify.

In Deep Learning, I have worked with Convolutional Neural Networks (CNNs) and applied Data Augmentation and Image Processing using TensorFlow and Keras. For deploying deep learning models, I have used TensorFlow Lite (TFLite).

For application authentication, I am familiar with JWT and OAuth.

For APIs, I have worked with Flask-based APIs, FastAPI, and Postman for testing.

For logging functionality, I have used Python's logging module.

For python backened website I am familiar with Django.  
  
For GenAI I had worked on how to use LLM MODELS APIs

Hi,  
  
This is Umakshi, Welcome to my world.  
  
I am an Aspiring AI Engineer, as we all know AI is shaping the future and its fascinating too. So I am on my journey to explore more about it.  
  
I am currently working as an AI SCIENTIST at iAssist Innovations Labs. A company which shaping the Insurance industry by its AI solutions for end to end claim processing. AS we know Maintaining a product is as important as making it. In Iassist I spearheading the enhancement of a product of ICICI Lombard called mICRa. I took the charge of reporting module there and analyzes the data to enchance the users experience. I Converted its monolithic architecture into micro-services as per the need and Learned how to develop FRS and BDS for the MIBL clients as per client requirement.

Before joining iassist, I did internship as data scientist at Exposys Data labs, ML/AI INtern at CSIR-CSIO Chandigarh , Andriod developer for a startup named OOHR INNOVATIONS and last but not the least my first intership as SUBJECT MATTER EXPERT OF mathematics at Shadovein.

In different different internships I learned different different things but one thing is common work ethics, quality is better than the quantity.

Talking about my educational background, I completed my Electronics and communiation degree from UIET of kurukshetra univeristy with 8.54 as my Overall CGPA and in my senior secondary eduction I had secured 93% from Non medical in CBSE borads.  
  
I born and brought up in Karnal and currently living in Kurukshetra.  
  
During my college I was the Content writer lead at E-CELL and had organised mega fests like EXCELSIOR, KAVACH, hackathons

According to me - my major strengths are my learning adaptability, curiosity, I can work under the pressure, Prompt engineering and make use of AI , Python

Hi,

This is Umakshi,

I am an **AI Scientist**, passionate about exploring the limitless possibilities of Artificial Intelligence. As we all know, AI is shaping the future and its fascinating too. So I am on my journey to explore more about it.

Currently, I am working as AI SCIENTIST at **iAssist Innovations Labs**, a company revolutionizing the insurance industry with AI-driven solutions for **end-to-end claim processing**. At iAssist, I spearheading the enhancement of **mICRa**, a product developed in collaboration with **ICICI Lombard**. I took charge of the **reporting module**, Every day, I get to analyze complex datasets, derive meaningful insights, and fine-tune AI-driven solution to enhance the user experience. Additionally, I played a key role in **converting its monolithic architecture into microservices**, making it more scalable and efficient. Along the way, I also learned how to develop **Functional Requirement Specifications (FRS) and Business Design Specifications (BDS)** to cater to MIBL clients' needs.

But my AI journey didn’t start here. It’s been a thrilling ride through multiple internships, each one shaping me into the professional I am today:  
🔹 **Data Scientist Intern at Exposys Data Labs** – Developed predictive models analyzing R&D, Administration, and Marketing Spend, achieving a 97.95% accuracy using Linear Regression.

🔹 **ML/AI Intern at CSIR-CSIO, Chandigarh** – Worked on an AI Car Assistant project . I worked on Data preprocessing here.  
🔹 **Android Developer at OOHR Innovations** – Contributed to the creation of a driver's app by preparing workflows and flowcharts, demonstrating effective app design strategies.

🔹 **Subject Matter Expert (Mathematics) at Shadovein** – Strengthened my problem-solving skills.

Each internship taught me something different, but one lesson remained constant: quality **matter more than quantity.**

Talking about my educational background, I hold a degree in **Electronics and Communication Engineering** from **UIET, Kurukshetra University**, with an **8.54 CGPA**. I secured **93% in my Senior Secondary Education (CBSE - Non-Medical stream).**

I was born and brought up in **Karnal** and currently reside in **Kurukshetra**.

Beyond academics, I was the **Content Writer Lead** at **E-CELL** in college and played a crucial role in organizing **mega fests like EXCELSIOR, KAVACH, and Hackathons, gaining hands-on experience in team coordination and large-scale event management.**

I believe my greatest strengths lie in my **learning adaptability, curiosity, ability to perform under pressure, and I am refining my skills in AI, Python, and prompt engineering.**

**I am** Excited to explore new challenges and opportunities in this journey! 🚀

 Hi,  
This is Umakshi,an Aspiring AI Engineer. As we all know, AI is shaping the future and its fascinating too. So I am on my journey to explore more about it.Currently, I am working as AI SCIENTIST at iAssist Innovations Labs, a company revolutionizing the insurance industry with AI-driven solutions for end-to-end claim processing. At iAssist,

I spearheading the enhancement of mICRa, a product developed in collaboration with ICICI Lombard. I took charge of the reporting module, Every day, I get to analyze complex datasets, derive meaningful insights, and fine-tune AI-driven solution to enhance the user experience. Additionally, I played a key role in converting its monolithic architecture into microservices, making it more scalable and efficient. Along the way, I also learned how to develop FRS and BDS according to the clients requirements.

But my AI journey didn’t start here. It’s been a thrilling ride through multiple internships, each one shaping me into the professional I am today. Before joining iAssist i worked as Data Scientist Intern at Exposys Data Labs where I Developed a predictive model based on their-problem statement of analyzing the impact of R&D, Administration, and Marketing Spend of a company on its Profit. I achieving a 97% accuracy using Linear Regression.

I worked as ML/AI Intern at CSIR-CSIO, Chandigarh where i Worked on an AI Car Assistant project and learned Data pre-processing here. I also worked as Android Developer in a startup named OOHR Innovations where I Contributed to the creation of a driver's app by preparing workflows and flowcharts, demonstrating effective app design strategies.

And last but not the least my first internship as Subject Matter Expert of Mathematics at Shadovein where i Strengthened my problem-solving skills.Each internship taught me something different, but one lesson remained constant: quality matter more than quantity.

Talking about my educational background, I hold a degree in Electronics and Communication Engineering from UIET of Kurukshetra University, with an overall CGPA of 8.54 and I secured 93% in my Senior Secondary Education from Non-Medical in CBSE boards.Beyond academics, I was the Content Writer Lead at E-CELL in college and played a crucial role in organizing mega fests like EXCELSIOR, and Hackathons like KAAVACH, where i gained hands-on experience in team coordination and large-scale event management.

I believe my greatest strengths lie in my learning adaptability, curiosity, ability to perform under pressure, and I am refining my skills in AI, Python, and prompt engineering.I am Excited to explore new challenges and opportunities in this journey!

***Play ai API user id - 22dhC2FB3QbJruc5xTZctRkO1142  
secret key - ak-5820bc01432f4521ab794de018f97236***

<https://app.play.ht/agents/agent/Umakshi-D4Mxb62CSen4dpcE7WX_M>

<script type="text/javascript" src="https://unpkg.com/@play-ai/agent-web-sdk@ht"></script>

<script type="text/javascript">

addEventListener("load", () => {

PlayAI.open('jIT8aLRd4EDtNtzEr6ACT');

});

</script>

Question Answers

Backend Framework Experience

\*

Q.1 : Do you have experience building a backend system with FastAPI? If yes, can you describe the projects briefly. If no, what other Python-based frameworks have you used?

No, I haven’t had the opportunity to work with FastAPI in a production environment yet. However, I do have experience with other Python-based web frameworks, particularly Flask and some exposure to Django.

At iAssist Innovations Labs, I primarily worked with Flask to develop and deploy backend services as part of the mICRa product for ICICI Lombard. My responsibilities included automating monitoring workflows, building RESTful APIs, and enhancing the reporting module by integrating Flask with Pandas. I also led the transition of a monolithic application into a microservices-based architecture, which significantly improved scalability and maintainability. This involved containerizing the services using Docker and implementing clean service-to-service communication.

Additionally, during my major academic project, I used Django to build API endpoints for a deep learning-based Ayurvedic herb classification system. The project involved integrating a CNN model with a Django backend to handle real-time inference and data management.

While I haven’t worked directly with FastAPI, I am confident that my experience with Flask and Django has given me a strong foundation in backend development. I’m eager to work with FastAPI, especially given its performance benefits and support for asynchronous operations.

LLM Backend Integration

\*

What challenges did you face integrating LLMs with your backend systems? How did you solve these challenges?

So while integrating LLMs in my personal projects, one of the key challenges I faced was ensuring that the model responded accurately and contextually, especially when I was building a personalized conversational AI chatbot using my own portfolio data. The idea was to help simulate recruiter interviews, basically, a chatbot that could answer common recruiter questions based on my actual work and experiences.

However, I noticed that despite providing a clear database and context, the chatbot would sometimes default to generic or standard responses almost like it was pulling from pre-trained public data rather than my custom input. For example, when asked about a project I worked on, it would sometimes give a textbook-style explanation instead of talking about what I actually did.

I realized this happened mostly due to context management issues and poor grounding of responses, especially when the retrieval component failed or wasn't fine-tuned properly. To solve this, I worked on improving the prompt structure, tweaking how data was embedded and retrieved, and also added guardrails to push it toward more relevant and personalized answers. It was a great learning experience in understanding the limits of LLMs when it comes to personalization and the importance of strong data grounding.

On the backend side, using Flask, I built the APIs to serve this bot and handled routing, embedding, and response generation logic modularly. My past work at iAssist, especially in converting monoliths into microservices, helped me design a clean and scalable architecture even in these experimental builds.

AI Orchestration Frameworks

\*

Which AI orchestration frameworks (like LangChain, LangGraph, LlamaIndex, etc.) have you used? What were the strengths and limitations you encountered?

I’ve worked with LangChain and LangGraph mainly during my personal projects while exploring agentic AI and LLM orchestration.

LangChain was my starting point. It’s great for chaining prompts, tools, and memory together - very modular. I used it while building a conversational AI agent that could answer recruiter-like questions from a personalized database. It worked well, but the complexity grew quickly. One challenge was that I had to manually describe the entire flow, step by step, which made debugging and scalability a bit tricky—especially when the conversation context wasn’t getting passed properly.

Then I explored LangGraph, and I really liked how it handled state and transitions. It’s more structured and declarative. One big advantage was that LangGraph already defines the flow based on the input, so I didn’t have to manually hardcode each step like in LangChain. This made the orchestration smoother and easier to visualize. It also helped when I wanted to maintain different states in longer conversations.

That said, LangGraph is still newer, so the learning curve was a bit higher, and there’s less community support or templates available. But it gave me more control and clarity, especially for multi-step reasoning tasks.

Overall, both helped me understand how to structure LLM applications better, even though I haven’t yet used them in production at iAssist.

LLM Observability

\*

What LLM observability tools or approaches have you used (for example, LangFuse, LangSmith, etc.)? How did you leverage them to improve your applications?

So at a production level, I haven’t worked with LangSmith, but I’ve explored it in my personal projects out of curiosity while experimenting with frameworks like LangChain and LangGraph. I came across LangSmith while learning how to better observe and debug LLM workflows,especially for tracing prompt chains and understanding how the model is responding at each step.

Most of my understanding came from official documentation, tutorials, and trial-and-error while building small prototype apps. What I found really helpful was how LangSmith allows you to track the entire flow, visualize prompt execution, and catch where things might go wrong, something that’s really useful when working on multi-step agent-based systems.

I haven’t used it in a professional setting yet, but based on my personal work, I see a lot of potential in using tools like this to improve reliability and transparency in LLM applications.

RAG Systems

\*

Can you walk through how you designed a Retrieval Augmented Generation (RAG) system? What indexing strategy and retrieval methods did you use, and how did you evaluate its effectiveness?

Yes, I’ve designed a RAG system in a personal project where I built a conversational AI chatbot that answers recruiter-related questions using my own structured data. I used FAISS as the vector store and OpenAI embeddings for converting chunks into dense vectors. For chunking, I experimented with different sizes (200–400 tokens) and overlap to maintain context, and added metadata filters to help with better relevance in retrieval.

Initially, I faced issues where the bot would return generic or unrelated answers even though the data was present. To fix this, I refined my chunking strategy and used hybrid search (keyword + vector) to improve accuracy. For evaluation, I did manual testing with 15–20 curated recruiter-style questions and tracked precision and relevance based on retrieved source chunks and generated answers.

This helped me understand the importance of clean data, chunk design, and prompt structure in RAG effectiveness.

LLM Adaptation Techniques

\*

Describe your experience with LLM adaptation techniques from parameter-efficient fine-tuning to full pre- training. What approaches have you found most practical for different use cases?

I haven’t worked on full pre-training of LLMs due to the heavy computational resources it requires, but I’ve explored parameter-efficient fine-tuning techniques like LoRA and prompt tuning in my personal projects. These methods were really practical for customizing base models without needing large infrastructure.

For example, I used LoRA to fine-tune a model for a personalized conversational chatbot using my own dataset (based on domain-specific recruiter FAQs). This allowed me to get more relevant responses without retraining the entire model. I found these methods to be more resource-friendly and flexible, especially for quick iterations and experiments.

In real-world use cases, these adaptation techniques are helpful when you want to retain the general knowledge of the base model but still customize it to your specific needs—like internal document Q&A or support bots. So overall, I’ve found parameter-efficient tuning to be the most practical in both experimentation and lightweight deployments.

End-to-End AI Project

\*

Describe an end-to-end AI project you built. What were the biggest technical challenges and how did you overcome them?

I worked on an end-to-end AI project where I built a personalized conversational AI chatbot using my own data to answer recruiter-specific questions. The biggest challenge was handling irrelevant or generic responses—even when the data was present, the LLM would sometimes default to standard answers. I improved this by refining the chunking logic during vectorization, adding metadata filters, and improving prompts.

Another technical challenge was data preprocessing and tuning the retrieval component in the RAG setup. I tried different chunk sizes and overlap strategies to get better context matching. I also dealt with latency issues, especially during retrieval + generation, and optimized by switching to lighter models for testing and using external APIs for production-level performance.

This project gave me practical experience in handling RAG pipelines, prompt tuning, and overcoming LLM-specific limitations with retrieval and latency.

Deployment & Scaling

\*

Beyond containerisation, what strategies have you used for deploying and scaling AI models in production?

Beyond containerization, I’ve focused a lot on optimizing the models themselves and building efficient pipelines to deploy and scale AI solutions. For example, at iAssist, I worked on converting a monolithic app into microservices, which helped improve scalability and made it easier to update parts of the system independently.

I also used automation to streamline model training and deployment—setting up scripts and workflows that ensure models are retrained and redeployed efficiently without manual intervention. This kind of pipeline helps maintain consistent performance as new data comes in.

Additionally, I pay attention to data preprocessing and feature engineering to make sure the models get high-quality input, which reduces the need for heavy compute resources during inference. Monitoring system performance and using lightweight APIs (like Flask) for serving models also helps keep response times low and the system scalable.

Overall, my approach combines modular architecture, automation, and careful optimization to ensure AI models run smoothly and can handle increasing loads in production.

Model Context Protocol

\*

The Model Context Protocol enables standardized integration between LLMs and tools. Have you worked with MCP or similar integration frameworks? How would you approach building an MCP server for [specific domain/use case]?

I haven’t worked directly with the Model Context Protocol (MCP) itself, but I’m familiar with the concept of standardized integration frameworks between LLMs and external tools. In my projects, I’ve focused on connecting AI models with backend services through APIs and microservices, which is somewhat similar in spirit.

If I were to build an MCP server for a specific domain, say for an insurance claims processing system, I would start by clearly defining the types of context and data the LLM needs access to—like claim details, policy info, and customer data. Then, I’d design APIs that securely expose this data and enable the LLM to query or update information as needed.

I’d also ensure the system can handle session management and context persistence efficiently, so the LLM has relevant history during interactions. Lastly, I’d emphasize modularity and scalability, likely using microservices and containerization (Docker), to allow easy updates and maintenance.

Overall, my approach would focus on creating a smooth, secure flow of information between the LLM and the backend to enable intelligent, context-aware responses in the application.

Prompt Engineering

\*

How do you approach prompt engineering? Can you describe techniques you've used to improve reliability and address hallucinations?

When I work on prompt engineering, I try to make the prompts very clear and simple so that the model understands exactly what I want. I start with a basic prompt and test how the model responds, then I make small changes to improve it. I use techniques like few-shot prompting, where I also give it examples when needed so it can follow the pattern, which helps in getting more accurate results.

To avoid wrong or made-up answers (hallucinations), I give strict instructions in the prompt and try to provide all the needed information within the context. If the model doesn’t get the right answer, I rephrase the question or guide it better.

One technique I found really useful is chain of thought prompting. This means asking the model to think step-by-step. For example, instead of just asking a direct question, I would say “let’s solve this step by step” and break the problem into small parts. This helps the model reason better and improves the accuracy of the answer.

Overall, I focus on making the prompt clear, giving context or examples, and guiding the model to think in steps when needed.

Multi-Modal Applications

\*

Have you built applications that incorporate multiple modalities (text, images, audio)? What unique challenges did you face when working with multi-modal models?

I haven’t worked directly on multi-modal applications in production yet, but I do have a basic understanding of how they function. I’ve explored some small personal projects combining text and images, like using OCR for image-to-text tasks or experimenting with image captioning using pre-trained models. One thing I noticed is that handling and preprocessing image or audio data is more complex compared to plain text. Also, aligning the outputs of different modalities (like matching an image with the right text response) can be tricky and often needs a lot of fine-tuning. I’d definitely love to work more with multi-modal systems in the future to get hands-on experience with real-world challenges.