



Avowal Data Systems

AI Automation

GENERATIVE AI PROGRAM

How Machines Create. We Empower.
The Future is Generative.

Trainer
Raju K



INDUSTRY RELEVANT
CURRICULUM



HANDS-ON
PROJECTS



REAL-WORLD
APPLICATIONS



CERTIFICATION
OF COMPLETION



ABOUT THE TRAINER

Raju Kolikapogu

Technical Architect

With over 20 years of experience, I specialize in architecting projects end-to-end and mentoring teams to success. I am also an expert in providing training for both individuals and corporate teams.



MY SKILL SET:



AI Automation:

Master n8n, Make.com, and Generative AI tools—langchain, RAG to streamline your workflows and boost efficiency.



Chatbots:

Master the art of building advanced chatbots w/o voice using VAPI, Voiceflow, BotPress, and Dialogflow to create seamless and engaging user interactions.



Data Analytics:

Harness the power of AI, Machine Learning, and Python to extract meaningful insights from data and drive business decisions, leveraging tools like Airflow, Spark, and Kubeflow for large-scale data handling.



20+
Years of
Experience



Trained
Individuals &
Corporate Teams

Foundations of Text Generation

1

Welcome to Generative AI!



- What is an LLM?
- How does AI predict the next word like your phone's autocomplete?

2

Your First API Call



- How do programs talk to AI models across the internet?
- Understanding API keys.

3

Local AI with Ollama



- How do we bring powerful models onto our own computers?
- Running AI completely offline.

4

Coding with Local Models



- How to use Python to command a model that is running on your hard drive.

5

Prompt Engineering Basics



- What is Prompt Engineering?
- How changing instructions completely alters the AI's output.

6

System vs. User Prompts



- Understanding the difference between the hidden instructions (System) and user questions (User).

7

Temperature & Creativity



- What is Temperature?
- Controlling how random, creative, or strict the AI's word choices are.

Introduction to LangChain

1 Introduction to LangChain



- What is LangChain?
- Why do software engineers use frameworks instead of raw API text strings?

2 Dynamic Prompt Templates



- Moving away from hardcoded strings to dynamic, reusable prompt blueprints.

3 Your First LangChain Chain



- Connecting a Prompt Template directly to an LLM model using the `|` (pipe) operator.



Advanced Chaining & Memory

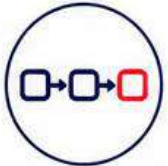
1



Sequential Chains (Part 1)

- What is sequential chaining? Feeding the output of one AI generation step into a second step.

2



Sequential Chains (Part 2)

- Coding multiple steps together to handle complex multi-stage text tasks automatically.

3



The Problem with AI Memory

- Why do LLMs naturally "forget" past messages? Understanding the stateless nature of standard web APIs.

4



Adding Conversation History

- How to append past questions and answers to the current prompt payload so the AI remembers context.

5



Conversation Window Buffers

- How to prevent chat histories from becoming too long and expensive by only keeping the last 'K' messages.

Structured Data Extraction

1



Why Structured Output Matters

- Standard paragraphs of text are hard for databases to read. Why we need AI to output clean data formats.

2



Introduction to Pydantic

- Using Python's Pydantic library to define exact data structures and rules that variables must follow.

3



Forcing JSON from an LLM

- Instructing the text generator to strictly follow your Pydantic schema instead of typing casual text.

Retrieval-Augmented Generation (RAG)

1



What is RAG?

- LLMs don't know your private data. Introduction to Retrieval-Augmented Generation (RAG) to stop hallucinations.

2



Loading Documents into AI

- How to pull text content out of standard office files like `.txt` or `.pdf` files using Python.

3



The Art of Text Splitting

- Why we can't throw a whole book at an AI at once. Breaking long documents into smaller, bite-sized "chunks".

4



What are Embeddings?

- How computers turn words into lists of numbers (vectors) to measure how similar two sentences are.

5



Your First Vector Database

- What is a Vector DB? A special filing cabinet built to look up text based on its meaning, not exact keywords.

Retrieval-Augmented Generation (RAG)

6



Saving Text Chunks to DB

- Taking your split document paragraphs, turning them into embeddings, and saving them to your local database.

7



Searching by Meaning

- Asking your vector database a question and having it pull up the exact paragraph containing the answer.

8



Building the RAG Pipeline

- The final formula: User question -> Vector DB search -> Relevant paragraph -> LLM generates the answer.

9



Adding Memory to RAG

- Combining your document-reading pipeline with conversation history so users can ask follow-up questions.

Web Deployment & Final Showcase

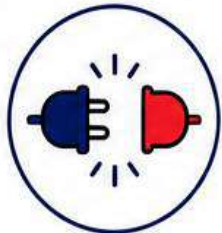
1



Building a User Interface

- Introduction to building clean, interactive web applications for your AI scripts without writing HTML/CSS.

2



Connecting the App to the Web UI

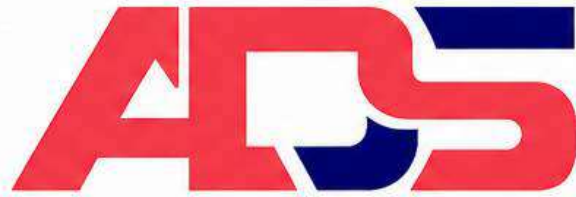
- Moving your final Python AI code out of the black terminal window and wiring it straight into your Web UI buttons.

3



Grand Finale Project Demo

- Reviewing your work, testing edge cases, cleaning up code comments, and celebrating your production-ready AI.



Avowal Data Systems

AI Automation

Contact Us



● **Contact:** (+91) 040 4511 5274



● **WhatsApp:** (+91) 78420 76060



● **Email:** info@avowaldatasystems.com



● **Website:** www.avowaldatasystems.com



● **Address:** Begumpet, Hyderabad, Telangana, India