

## Generative AI

### Introduction to Generative AI

- **Overview of Generative AI:** Define generative AI, its significance, and applications.
- **History of Language Models:** From symbolic NLP to neural NLP and state-of-the-art models.

### Understanding LLMs

- **What Are LLMs?**
  - Basics of LLMs, their applications, and how they work.
  - Comparing human thought with LLM processing.
- **How LLMs Work**
  - Tokenization, patterns, and repetitions.
- **Model Behavior**
  - Hallucinations in LLMs.
  - Temperature and probabilities in output generation.

### Prompt Engineering

- **Basics of Prompt Engineering:** Crafting effective prompts for desired outputs.
- **Techniques and Strategies**

### Advanced Prompts

#### Application Development with LLMs

- **Designing LLM Applications**
  - Identifying user problems and converting them to model domains.
  - Using LLMs to complete prompts and transform back to user space.
- **Quality Evaluation**
  - Online and offline evaluation of LLM application quality.

- Reinforcement Learning from Human Feedback (RLHF).
  
- **Incorporating Frameworks and Tools**
  - **Programming Frameworks:**
    - **LangChain, LLMIndex:** Using open-source tools for building LLM applications.
    - **Anarchy:** Another open-source tool for development.
  - **GUI Frameworks:**
    - **Flowise:** Open-source GUI framework for LLM applications.
    - **Stack AI:** Commercial GUI framework for advanced applications.
  - **Monitoring Tools:**
    - **Autoblocks, Helicone, HoneyHive, LangSmith, Weights & Biases:** For monitoring and managing LLM applications.
  - **Caching Tools:**
    - **GPTCache, Redis:** For efficient data handling and storage.
  - **Validation Tools:**
    - **Guardrails AI, Rebuff:** Ensuring quality and reliability of LLM outputs.

## Introduction to LangChain

- **Overview**
- **Key Components of LangChain**
  - **Agents:** Execute tasks and actions based on model outputs.
  - **Chains:** Sequences of tasks or prompts executed in a defined order.
  - **Memory:** Enables context retention across interactions.
  - **Tools:** Various tools for data retrieval, transformation, and more.

## Advanced LangChain Features

- Agents, Multi Agents, RAG, Vector Databases

## Handling Hallucinations

- Strategies to mitigate and manage incorrect or nonsensical outputs.

## Optimizing Performance

- Techniques for improving efficiency and effectiveness of LLM applications.

## Deployment and Monitoring

- Best practices for deploying LangChain applications, including CI/CD pipelines and monitoring frameworks for observability and maintenance.

## Fine-tuning and Evaluating LLMs

- **Fine-tuning**
  - Fine-tuning models for specific tasks.
- **Model Evaluation**
  - Evaluating the performance and effectiveness of fine-tuned models.
- **Observability and Testing**
  - **Observability Frameworks:**
    - Implement monitoring to gain insights into model performance and behavior.
    - Use tools like **NVIDIA Guardrails, Weights & Biases, LangSmith, Helicone** for observability.
  - **Testing Strategies:**
    - Employ robust testing practices for LLM applications.
    - Tools like **LangSmith** for comprehensive testing and validation.

## Deployment and Orchestration

- **Deployment Strategies:**
  - Best practices for deploying LLM applications.

- Use tools like **LangServe** to manage model deployment and scaling.
- **Operationalization (LLMOPs):**
  - Implementing continuous integration and deployment (CI/CD) pipelines.
  - Utilize tools for automated deployment and scaling.

### **Reinforcement Learning and LLM-Powered Applications**

- **Reinforcement Learning with Human Feedback (RLHF)**
  - Implementing RLHF in LLMs for improved performance.
- **Model Optimization for Deployment**
  - Techniques for optimizing models for real-world applications.
- **Using LLMs in Applications**
  - Practical applications of LLMs in various domains.

