Faysal Aziz

Advanced Generative AI with Large Language Models (LLMs)

2-day highly intensive training outline

FA-0512 · 2-day

Agentic & Generative AI

Introduction

This course is a highly intensive, advanced training program designed for experienced developers to dive deep into the world of Generative AI, specifically focusing on Large Language Models (LLMs) such as Llama2, Claude 2, MPT-7B, and others. The training will cover not only the practical application of these models using Python but also the underlying mathematics and theory that enable their functionality. Additionally, we will explore tools like LangChain, Pinecone, Python-dotenv, and GPT wrappers, which are essential in deploying these AI models effectively.

Learning outcomes

By the end of this training, participants will be able to:

- Understand the mathematical and theoretical foundations of Generative AI and LLMs.
- Implement and manipulate open-source LLMs using Python for various applications.
- Integrate LLMs with retrieval-augmented generation (RAG) to create end-to-end solutions.
- Utilize LangChain for chaining together components of language model applications.
- Implement Pinecone for vector search in AI applications.
- Configure and manage environment variables securely using Python-dotenv.
- Employ GPT wrappers for efficient interaction with Generative AI models.
- Troubleshoot and optimize the performance of LLMs in different scenarios.

Prerequisites

- Proficiency in Python programming, including familiarity with Python's advanced features and standard libraries.
- Solid understanding of machine learning concepts, including neural networks, backpropagation, and loss functions.
- Basic knowledge of Natural Language Processing (NLP) and text preprocessing techniques.
- Experience with data handling and manipulation using libraries such as NumPy, Pandas, and PyTorch/TensorFlow.
- Familiarity with command-line tools and version control systems (e.g., Git).
- Basic understanding of web APIs and application integration.
- Comfort with mathematical concepts, including linear algebra, calculus, and probability theory.
- Access to a computer with high processing power, a good internet connection, and permissions to install various packages and software.

Training outline

1. Generative AI and LLMs Overview

1. Evolution of AI and the generative model landscape
2. The architecture of transformer-based models
3. Comparative analysis of Llama2, Claude 2, MPT-7B, etc.

2. Mathematical Foundations

1. Deep dive into the math of transformers: attention mechanisms
2. Probability, Perplexity, and Loss Functions
3. Optimization techniques and gradient descent variations

3. Setting Up the Development Environment

1. Python environment setup (virtual environments, Jupyter notebooks)
2. Introduction to Python-dotenv for environment variable management
3. Version control and collaborative development best practices

4. Working with Open Source LLMs

1. Downloading, installing, and configuring LLMs

2. Best practices for model fine-tuning and parameter adjustment
3. Advanced text generation techniques

5. Implementing RAG (Retrieval-Augmented Generation)

1. Understanding the RAG approach and its significance
2. Integrating databases and knowledge sources
3. Customizing RAG for specific use cases

6. LangChain for AI Application Chaining

1. Introduction to LangChain library
2. Building language model chains for complex tasks
3. Advanced feature exploration and custom component creation

7. Vector Search with Pinecone

1. Basics of vector search and its relevance to AI
2. Setting up and using Pinecone in AI applications
3. Indexing and querying vectors for large-scale data

8. Security and Ethics

1. Handling data securely in Generative AI
2. Ethical considerations in Generative AI development and deployment
3. Bias detection and mitigation strategies

9. GPT Wrappers and API Interaction

1. Overview of GPT wrappers for Python
2. Building and consuming REST APIs with LLMs
3. Error handling and rate limiting strategies

10. Deployment and Scaling

1. Deployment strategies for LLM-based applications
2. Scalability considerations and cloud integration
1. Monitoring and maintenance best practices

11. Case Studies and Hands-on Projects

1. Analysis of successful Generative AI projects
2. Collaborative project: building a sample end-to-end solution with LLMs and RAG
3. Performance evaluation and troubleshooting

This outline provides a structured pathway for highly experienced developers to not only grasp the intricate workings of LLMs and related technologies but also to apply this knowledge in creating sophisticated AI-driven solutions.

© 2026 Faysal Aziz · Kuala Lumpur