# Advanced GenAI Workshop

========================================================================

**Courseware:** Unofficial PPT

**Lab:** Open-source platform

========================================================================

**Pre-requisites:**

**Python:** Proficiency in Python and data libraries (e.g., NumPy, pandas).

**Machine Learning:** Basic understanding of ML concepts and algorithms.

**Deep Learning:** Knowledge of neural networks and frameworks (e.g., TensorFlow)

**NLP:** Familiarity with NLP tasks and techniques.

**Transformers:** Basic understanding of Transformer architecture and attention mechanisms.

**Open-Source Tools:** Experience with open-source platforms.

======================================================================== **Learning Outcomes:**

The Advanced GenAI Workshop provides a comprehensive exploration of cutting-edge Generative AI techniques. Participants will gain foundational knowledge in NLP tasks, word embeddings, and attention mechanisms, and understand Transformer architecture and its advantages. The course includes practical training with LangChain for building Retrieval Augmented Generation (RAG) and advanced QnA systems, utilizing tabular and SQL data. Additionally, attendees will learn fine-tuning techniques for optimizing large language models (LLMs) like Llama and Gemma, and apply quantization methods to enhance model efficiency. The workshop concludes with hands-on evaluation of LLMs using MLflow, ensuring effective tracking, management, and real-world performance assessment.

========================================================================**Module 01: Prerequisites to Transformer Architecture**

**Outcome:** Learn NLP basics, word embeddings, and attention mechanisms, including Self-Attention. Understand the limitations of RNNs and explore Transformer architecture with Multi-Head Attention, focusing on how Transformers handle long-range dependencies efficiently in NLP tasks.

- Overview of NLP tasks: Translation, Sentiment Analysis, etc.
- Word embeddings: Bag-of-Words, Word2Vec.
- Introduction to Self-Attention Mechanism
- Importance in enhancing Seq2Seq models.
- Encoder-Decoder Architecture.

- Limitations of RNN-based models (e.g., handling long-range dependencies).
- Self-Attention Basics.
- Difference from regular attention.
- Core concepts: Query, Key, Value; benefits over RNNs.
- Key components: Multi-Head Attention, Encoder-Decoder blocks.
- Advantages: Parallelization, better handling of dependencies.

**Module 02: Basic LLM Systems (RAG) using LangChain**

**Outcome:** Understand Retrieval Augmented Generation (RAG) and use LangChain to build LLM applications, chatbots, and retrieval systems. Develop practical skills in creating conversational RAG applications, enhancing knowledge in retrieval-based AI model building.

- Introduction to Retrieval Augmented Generation (RAG)
- Deepdive to LangChain
- Concept of Embedding, Retrieval, Chain and Agents using LangChain
- Build a Simple LLM Application using LangChain
- Build a Chatbot LangChain
- Build vector stores and retriever using LangChain
- Build an Agent LangChain
- Build a Retrieval Augmented Generation (RAG) Application using LangChain
- Build a Conversational RAG Application using LangChain

**Module 03: Advanced LLM Systems (QnA) using LangChain**

**Outcome:** Learn to differentiate RAG from QnA systems and build advanced QnA systems using LangChain. Gain hands-on experience with tabular and SQL data for complex question-answering tasks.

- Difference between RAG & Question Answering system
- Build a Question Answering system over Tabular Data using LangChain
- Build a Question/Answering system over SQL data using LangChain

**Module 04: Fine-tuning Techniques for Large Language Models**

**Outcome:** Explore fine-tuning techniques for optimizing LLMs. Learn quantization methods and their applications to models like Llama and Gemma, enhancing model efficiency for specialized tasks.

- Introduction to Quantization
- Optimization of model weights (data types)
- Modes of Quantization
- Fine tuning LLMs (Meta’s Llama / Alibaba’s Qwen / Google’s Gemma)

**Module 05: Evaluation of Large Language Models using MLflow**

**Outcome:** Learn to evaluate LLMs using MLflow, with hands-on experience in tracking, managing, and fine-tuning models, including evaluating Hugging Face models to ensure optimal real-world performance.

- Introduction to MLflow
- Build a machine learning model using MLflow
- LLM Evaluation using MLflow
- Evaluate a Hugging Face LLM

================================================================
======