



## Open Source Generative AI

- 4-Day Class
- Hands-on labs

### Course Overview

Open-Source Generative AI for the Enterprise is a comprehensive 4-day course that teaches practical applications for AI in the business environment. Design, and develop your own Transformer model GenerativeAI, ensuring data security for proprietary data. This course offers a combination of lectures and hands-on labs, providing participants with a solid understanding of AI concepts and the skills to design and implement AI solutions.

Throughout the course, you will learn about AI transformer-based architectures, the fundamentals of Python programming for AI models, and the deployment of open source Transformer models. You will also explore the importance of hardware requirements in AI performance, comparing different GPU architectures and understanding how to match AI requirements with suitable hardware. The course delves into training techniques, including back propagation, gradient descent, and various AI tasks such as classification, regression, and clustering.

You will gain practical experience through hands-on exercises with open source LLM (Language Learning Model) frameworks, allowing you to work with fine-tuned models and run workloads on different models to understand their strengths and weaknesses. Additionally, the course covers the conversion of model formats and provides in-depth exploration of AI programming environments like PyTorch+transformers and transformers' low-level interactive inspection.

Towards the end of the course, you will delve into advanced topics such as context extension through fine-tuning and quantization for specific application target environments. By the completion of the course, you will have the opportunity to earn an AI certification from Alta3 Research, further enhancing your credentials in the field of Artificial Intelligence. This course is ideal for Python Developers, DevSecOps Engineers, and Managers or Directors seeking a practical overview of AI and its practical application in the enterprise.

Note: Students are expected to already have basic Python skills.

- Access the classroom from anywhere via browser and internet
- Each participant will have access to a fully configured, GPU-accelerated server.
- Obtain hands-on experience with the most widely used, industry-standard software, tools, and frameworks.
- Learn to build deep learning and accelerated computing applications

### Who Should Attend

- Project Managers
- Architects
- CKA Developers
- Data Acquisition Specialists








### What You'll Learn

- Understand AI architecture, specifically the Transformer model.










- Describe the role of tokenization and word embeddings in AI processing.
- Train and optimize a Transformer model using PyTorch.
- Master advanced prompt engineering for model control.
- Install and use AI frameworks like Llama-2.
- Explore model quantization and fine-tuning.
- Understand different AI interaction modes (chat vs. instruct).
- Maximize AI Model Performance
- Compare hardware acceleration options (CPU vs. GPU).
- Apply strategies to maximize AI model performance.
- Write a real world AI Web Application

## Outline






### Deep Learning Intro

-  Lecture: What is Intelligence?
-  Lecture: Generative AI Unveiled
-  Lecture: The Transformer Model
-  Lecture: Feed Forward Neural Networks
-  Lecture + Lab: Tokenization
-  Lecture + Lab: Word embeddings
-  Lecture + Lab: Positional Encoding




### Build a Transformer Model from Scratch

-  Lecture: PyTorch
-  Lecture + Lab: Construct a Tensor from a Dataset
-  Lecture + Lab: Orchestrate Tensors in Blocks and Batches
-  Lecture + Lab: Initialize PyTorch Generator Function
-  Lecture + Lab: Train the Transformer Model
-  Lecture + Lab: Apply Positional Encoding and Self-Attention
-  Lecture + Lab: Attach the Feed Forward Neural Network
-  Lecture + Lab: Build the Decoder Block
-  Lecture + Lab: Transformer Model as Code


### Prompt Engineering








-  Lecture: Introduction to Prompt Engineering
-  Lecture + Lab: Getting Started with Bard
-  Lecture + Lab: Developing Basic Prompts
-  Lecture + Lab: Intermediate Prompts: Define Task/Inputs/Outputs/Constraints/Style
-  Lecture + Lab: Advanced Prompts: Chaining, Set Role, Feedback, Examples

### Hardware requirements

-  Lecture: GPUs role in AI performance (CPU vs GPU)
-  Lecture: Current GPUs and cost vs value
-  Lecture: Tensorcore vs older GPU architectures

### Pre-trained LLM

-  Lecture: A History of Neural Network Architectures
-  Lecture: Introduction to the LLaMa.cpp Interface
-  Lecture: Preparing A100 for Server Operations
-  Lecture + Lab: Operate LLaMa2 Models with LLaMa.cpp
-  Lecture + Lab: Selecting Quantization Level to Meet Performance and Perplexity Requirements

-  Lecture: Running the llama.cpp Package
-  Lecture + Lab: Llama interactive mode
-  Lecture + Lab: Persistent Context with Llama
-  Lecture + Lab: Constraining Output with Grammars
-  Lecture + Lab: Deploy Llama API Server
-  Lecture + Lab: Develop LLaMa Client Application
-  Lecture + Lab: Write a Real-World AI Application using the Llama API

#### Fine Tuning

-  Lecture + Lab: Using PyTorch to fine tune models
-  Lecture + Lab: Advanced Prompt Engineering Techniques

#### Testing and Pushing Limits

-  Lecture + Lab: Maximizing Model Limits

#### Prerequisites

- Python - PCEP Certification or Equivalent Experience
- Familiarity with Linux