Building LLM Applications with Prompt Engineering

Duration: 1 Day **Course Code: GK847008** **Delivery Method: Company Event**

Overview:

Very large deep neural networks (DNNs), whether applied to natural language processing (e.g., GPT-3), computer vision (e.g., huge Vision Transformers), or speech AI (e.g., Wave2Vec 2) have certain properties that set them apart from their smaller counterparts. As DNNs become larger and are trained on progressively larger datasets, they can adapt to new tasks with just a handful of training examples, accelerating the route toward general artificial intelligence. Training models that contain tens to hundreds of billions of parameters on vast datasets isn't trivial and requires a unique combination of AI, high-performance computing (HPC), and systems knowledge.

Company Events

These events can be delivered exclusively for your company at our locations or yours, specifically for your delegates and your needs. The Company Events can be tailored or standard course deliveries.

Objectives:

- Train neural networks across multiple servers
- Use techniques such as activation checkpointing, gradient accumulation, and various forms of model parallelism to overcome the challenges associated with large-model memory footprint
- Capture and understand training performance characteristics to optimize model architecture
- Deploy very large multi-GPU models to production using NVIDIA Triton Inference Server

Content:

Module 1: Introduction

- Orient to the main workshop topics, schedule and prerequisites.
- Learn why prompt engineering is core to interacting with Large Language Models (LLMs).
- Discuss how prompt engineering can be used to develop many classes of LLM-based applications.
- Learn about NVIDIA LLM NIM, used to deploy the Llama 3.1 LLM used in the workshop.

Module 2: Introduction to Prompting

- Get familiar with the workshop environment.
- Create and view responses from your first prompts using the OpenAI API, and LangChain.
- Learn how to stream LLM responses, and send LLMs prompts in batches, comparing differences in performance.
- Begin practicing the process of iterative prompt development.
- Create and use your first prompt templates.
- Do a mini project where to perform a combination of analysis and generative tasks on a batch of inputs.

Module 3: LangChain Expression Language (LCEL), Runnables, and Chains

- Learn about LangChain runnables, and the ability to compose them into chains using LangChain Expression Language (LCEL).
- Write custom functions and convert them into runnables that can be included in LangChain chains.
- Compose multiple LCEL chains into a single larger application chain.
- Exploit opportunities for parallel work by composing parallel LCEL chains.
- Do a mini project where to perform a combination of analysis and generative tasks on a batch of inputs using LCEL and parallel execution.

Module 4: Prompting With Messages

- Learn about two of the core chat message types, human and AI messages, and how to use them explicitly in application code.
- Provide chat models with instructive examples by way of a technique called few-shot prompting.
- Work explicitly with the system message, which will allow you to define an overarching persona and role for your chat models.
- Use chain-of-thought prompting to augment your LLMs ability to perform tasks requiring complex reasoning.
- Manage messages to retain conversation history and enable chatbot functionality.
- Do a mini-project where you build a simple yet flexible chatbot application capable of assuming a variety of roles.

Module 5: Structured Output

- Explore some basic methods for using LLMs to generate structured data in batch for downstream use.
- Generate structured output through a combination of Pydantic classes and LangChain's `JsonOutputParser`.
- Learn how to extract data and tag it as you specify out of long form text.
- Do a mini-project where you use structured data generation techniques to perform data extraction and document tagging on an unstructured text document.

Module 6: Tool Use and Agents

- Create LLM-external functionality called tools, and make your LLM aware of their availability for use.
- Create an agent capable of reasoning about when tool use is appropriate, and integrating the result of tool use into its responses.
- Do a mini-project where you create an LLM agent capable of utilizing external API calls to augment its responses with real-time data.

Module 7: Assessment and Final Review

- Review key learnings and answer questions.
- Earn a certificate of competency for the workshop.
- Complete the workshop survey.
- Get recommendations for the next steps to take in your learning journey.

Further Information:

For More information, or to book your course, please call us on 0800/84.009

info@globalknowledge.be

www.globalknowledge.com/en-be/