TITAN
CONSULTING
INSIGHTS

GENERATIVE AI

INTRODUCTION, THINGS TO CONSIDER, AND BUSINESS CASES PART 2 OF 4

by Keith Johnson, Practice Manager, Titan Consulting

As a continuation of our Generative AI discussion, Part 2 in our series will discuss Large Language Models (LLMs) and two components of LLMs -- Fine-tuning and Prompt Engineering. Large Language Models (LLMs) have revolutionized the field of natural language processing, offering powerful capabilities for understanding and generating human-like text. These models, built on vast amounts of data and complex neural network architectures, have demonstrated remarkable performance in tasks such as translation, summarization, and content creation. However, to fully harness their potential for specific applications, two critical techniques come into play: Fine-tuning and Prompt Engineering. Fine-tuning involves adapting a pre-trained model to a particular task or domain by training it on specialized data, enabling more accurate and contextually relevant outputs. Prompt Engineering, on the other hand, focuses on optimizing how inputs are structured to guide the model's responses effectively. Together, these approaches enable precise control over LLMs' behavior, expanding their utility across diverse industries. This white paper delves into the fundamentals of Fine-tuning and Prompt Engineering, exploring techniques, benefits, and practical implications in deploying LLMs for targeted use cases.

— *Warren Norris, Managing Partner*

Generative AI – Large Language Models (LLMs)

Large Language Models (LLMs) are AI models specifically designed to understand and generate human text. These models excel in handling words, grammar, sentences, and context with high accuracy. Unlike Generative AI, which encompasses the generation of text, images, audio, video, and code, LLMs focus is exclusively on text.

Key Features of LLMs:

- Trained on massive amounts of data.
- Utilize large neural networks called transformers, with billions of parameters that enhance their text understanding and generation abilities.
- Fine-tuning is available and allows LLMs to perform specific tasks with greater accuracy.

Use Cases for LLMs:

- Text generation for marketing and advertising.
- Chatbots and virtual assistants for user support and interactions.
- Language translation.
- Text summarization, providing detailed and accurate summaries of long documents.
- Q&A to ask a question and get an answer.

TITAN
CONSULTING

Generative AI – Fine-tuning

Fine-tuning a LLM is the process of adapting a pre-trained model *to perform specific tasks or to cater to a particular domain more effectively.*

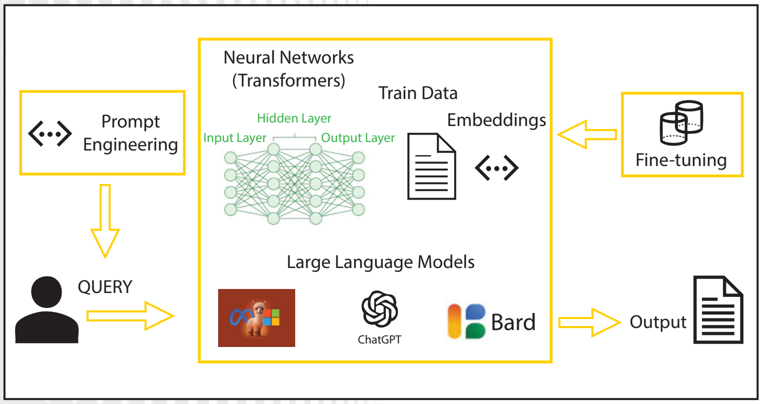
What does that mean? It is where the user adjusts a pre-trained LLM to do a more specific task.

How do you do this? We do this with *a more focused dataset.* (Ex: "If you want information back from ChatGPT that is specific to you, say your healthcare information that isn't available in the data that has been fed to ChatGPT, then you can input your data into ChatGPT for better results.")

Fine-tuning involves adjusting an existing, pre-trained model rather than creating something from scratch. It *builds on a foundation model* that has already been trained on a large volume of data, and the *process involves tweaking this model with additional data* to better suit specific results.

LLM Summary:

- User queries with text generating models such as ChatGPT, Bard, or Llama.
- The brain behind the model is called the neural network, more precisely, the transformer.
- These models are trained using large volumes of data, and they convert text into embeddings to understand their semantics, meaning, and context.
- How we can improve on these models to get better results:
  - Prompt Engineering – simply a way to frame better questions, expand on questions, explain background.
  - Fine-tuning – train the model more using a specific data set.



Generative AI – Prompt Engineering

Prompt Engineering is the process of crafting well-defined and structured input queries to interact with AI systems to generate accurate and relevant responses. It is a specific question, command, or input that you provide to an AI system to request a particular response, information, or action.
Remember, ChatGPT is like talking to a real person – if you ask clear, specific, and accurate questions in the correct context, you will get better answers.

Best Practices for Prompt Engineering:

- Clearly convey for the desired response.
- Provide context or background information – set some ground rules.
- Balance simplicity and complexity.
- Perfect through iterative and refinement.

Examples of Prompts:

- Tell ChatGPT to summarize a research paper or write a short story summary using bullet points.
- Tell Dall-E to generate images of a yellow banana.
- Tell GitHub Copilot to write a Python program snippet.

**Prompt Engineering is an Art...
Keep playing with it.**

Generative AI – Embeddings

In the LLM Summary above, we mentioned Embeddings. Embeddings are technical in nature, so without getting into too many technical details, we need to give a quick explanation of Embeddings.

Machines do not understand text, they only understand numbers. Embeddings are numerical representations of text. They are essential for AI models to understand and work with human language.

In a LLM, the neural network generates Embeddings, which are numerical representations of words that capture their meaning, context, and relationships. Each word is broken down into Tokens, which are then combined through a process called Chunking. The neural network uses its training on billions of data points to accurately generate these Embeddings. The transformer model understands the meaning of these numbers, and when generating sentences, ChatGPT constructs them one word at a time based on this learned knowledge.