(TR-103) PROMPT ENGINEERING –

Training Day 3 Report:

What is a Large Language Model (LLM)?

- A Large Language Model (LLM) is a type of AI trained to understand and generate human-like text.
- It uses vast datasets (books, websites, conversations) to learn patterns in language.
- Example: ChatGPT, Google Gemini, Claude, LlaMA

Key Terms in Large Language Models:

1. **Token:**
    - A token is a small piece of text that the model reads and processes.
    - It can be a full word, part of a word, or even a punctuation mark.
    - Large Language Models don't process sentences directly—they break them down into tokens first.
2. **Parameter:**
    - A parameter is like a setting or switch inside the model that it learns and adjusts during training.
    - These parameters help the model decide what the next word should be when it is generating text.
    - The more parameters a model has, the more knowledge and accuracy it can achieve.
3. **Prompt:**
    - A prompt is the input or question that you give to the model.
    - The model then uses the prompt to understand what kind of response it needs to generate.
4. **Fine-Tuning:**
    - Fine-tuning means taking a model that's already trained on general data and training it again on specific data to make it more suitable for a particular task or field.
5. **Inference:**
    - Inference is the stage where the trained model is used to make predictions or generate responses based on input prompts.
    - It happens after the training is complete.

What is LLM Architecture?

- The architecture of a Large Language Model (LLM) defines how the model is structured internally—how it processes input text, learns patterns, and generates meaningful output.
- Almost all modern LLMs, such as GPT, BERT, and Gemini, are built upon a foundation called the Transformer architecture, which allows them to handle large amounts of text efficiently and understand context better than older models.

Core Components of LLM Architecture:

1) **Input Text:**
   The model starts with a text input from the user.
2) **Tokenizer:**
   The input text is broken down into tokens—small units such as words or subwords.
3) **Embedding Layer:**
   - Each token ID is converted into a vector using an embedding layer.
   - Additionally, positional encoding is added to each token so the model knows the order of the words .
   - Output: A sequence of vectors representing both the meaning and position of each word.
4) **Transformer Blocks:**
   This is the core of LLMs. Each token passes through multiple transformer blocks, and each block contains:
   - Multi-Head Self-Attention: Helps the model focus on important words in the sentence
   - Feed Forward Neural Network (FFNN): A small neural network for deeper understanding of token meanings.
   - Add & Normalize: Keeps training stable and retains important information using residual connections and layer normalization.

   This structure is repeated many times.
5) **Output Layer:**

   After processing through the transformer blocks, the model predicts the next most likely token based on the given input and produces the output.

Training Large Language Models (LLMs): Behind the Scenes:

Large Language Models like GPT, Gemini, and Claude are developed through a multi-phase training process designed to help them understand language, perform specific tasks, and align with human preferences. The three major phases involved are:

1. **Pretraining:**
   Objective:
   - To enable the model to learn general language patterns, grammar, knowledge, and reasoning from large-scale datasets.

   How it works:
   - The model is trained on vast amounts of publicly available text such as books, websites, articles, and forums.
   - It learns to predict the next token in a sentence based on the previous ones (known as next-token prediction).
   - This phase is unsupervised, meaning it does not require manually labeled data.
2. **Fine-Tuning**
   Objective:
   - To specialize the pretrained model for specific use-cases, industries, or tasks.

   How it works:
   - The pretrained model is further trained on task-specific datasets .
   - This phase uses supervised learning, where correct input-output pairs are provided.
   - Fine-tuning improves the model's performance in targeted domains and makes it more useful in practical applications.
3. **Reinforcement Learning with Human Feedback (RLHF)**
   Objective:
   - To align the model's responses with human preferences, safety standards, and ethical guidelines.

   How it works:
   - The model generates multiple possible outputs for a prompt.
   - Human reviewers rank the responses based on clarity, helpfulness, and safety.
   - A reward model is trained to understand what humans prefer.
   - The main model is updated using reinforcement learning algorithms to improve its behavior over time.

Applications of LLMS:

- **Chatbots:** 24/7 customer support
- **Education:** Personalized tutoring
- **Healthcare:** Summarizing records
- **Legal:** Drafting contracts
- **Writing:** Blogs, poetry, code

Limitations of LLMS:

- **Hallucinations:** May generate false info.
- **Bias:** Reflects biases in its training data.
- **No real understanding:** Doesn't "think".
- **Context length:** Limited memory.

Ethical Concerns:

- Misinformation Generation
- Data Privacy
- Deepfakes and Impersonation
- AI Responsibility

Case Study: Microsoft's Tay chatbot went rogue in 24 hours due to biased input.