What is LLM?

Large Language Model

They are trained on massive text data

They are language models made of neural networks

They can understand human language and generate human language

A large language model is instance of Foundation model

GPT-4 , Claude , Gemini, Deepseek

How LLM works?

It works in different steps :

1. Training: LLM's are trained on large text data from Wikipedia, websites, social media posts, research papers, conversations etc.
   LLM prhta or seekhta hai.
   Imagine a person reading 1000's of new books articles blogs on daily basis. Us insaan ka dimaag language ke patterns seekh leta hai — jaise kis word ke baad kya word aata hai, ya koi sentence kis mood mein likha gaya hai.
   LLM bhi isi tarah patterns aur meanings seekhta hai — but much faster aur zyada detail mein.

2. Neural networks: Imagine them as brain that is inside LLM, similar to human brain and process same like as human brain.

3. Tokenization: As we all know computers only understand numbers. So, when we give any sentence to model it breaks that sentence into different token
   For example: I give a sentence "My name is Fariha"
   So, the model first breaks it into token or small words like:
   "My", "name", "is", "Fariha"
   Then assign a number to each token, then give that numbers to neural network.

4. Prediction: Model predict the next word based on the words given.It predicts based on the pattern it sees continuously during training.
   For example: If i ask : “The capital of Pakistan is.....”
   The higer chances are that model will give Islamabad , kyun ke usne training mein yeh pattern baar baar dekha hai.

5. Embedding: This assigning of numbers to token is embedding.
   Embedding is the process of converting words or tokens into **meaningful vectors (numbers)** that a model can understand.
   As we know computer can’t understand normal words like name , Pakistan etc, they only understand numbers. But simple numbers like 1, 2, 3 for each word don't capture **meaning**.
   So instead of assigning random numbers, LLMs use **embeddings** to represent each word as a **vector in a multi-dimensional space**.

6. Positional Encoding: Since LLMs (like transformers) **don't understand the order** of tokens by default, we add **positional encoding** to tell the model
   Let’s say:
   “My name is Fariha”
   Model needs to know:

- “My” is at position 1
- “name” is at position 2
- and so on...

So we add a **positional vector** to each word’s embedding.

What are Transformers?

A **Transformer** is a type of **deep learning model architecture** that is especially designed to **handle language** (text), and sometimes even images, code, and more.

Before transformers, models had problems like:

- Forgetting earlier words
- Being slow in training
- Not understanding long-range context well

Transformers solved all that by introducing a new idea: **Self-Attention**

Key Concepts In Transformers:

1. **Input: Token + Embedding + Position**

- Like before, input text is **tokenized**
- Each token is converted into a **vector (embedding)**
- Then we add **positional encoding** to know the word's position

2. **2. Self-Attention – "Kis Word ko Kitna Dhyan Dena Hai"**

- This is the magic inside Transformers.
- Imagine you're reading this sentence:

"Ali went to the bank to deposit money."

- The word **"bank"** can mean:
  A financial place
  A river bank
- To understand the correct meaning, the model **attends to nearby words** — like "deposit" or "money" — and realizes:
  "Oh! This is a financial bank."
- Model har lafz ke aas paas ke words ko dekhta hai aur decide karta hai:
  "Mujhe kis word par zyada focus karna chahiye?"
  This process is called **self-attention**.

3. **Multi-Head Attention**

- Instead of focusing on just **one kind** of relationship, the transformer uses **multiple heads** to focus on different relationships in parallel.
- **Socho 8 log aik sentence ko alag nazar se dekh rahe hain** — kisi ne grammar pe dhyan diya, kisi ne mood pe, kisi ne context pe.
  Then sabki analysis combine hoti hai.

4. **Feedforward Layers**
   After self-attention, the outputs are passed through **normal neural layers** to further process and refine the meaning.

5. **Stacked Layers**

Transformers have **many layers** — each learning deeper and more abstract patterns.

For example:

- First layers learn grammar
- Later layers learn intent or mood
- Final layers make predictions