**Matthew Ayodele** Week 1 & 2: Understand Large Language Model

Language models are advanced machine learning models designed to generate human-like text. They are trained on a wide range of texts from different sources. ChatGPT serves as an example of LLMs, and we will use it as a case study in this note.

A pre-trained Large Language Model is basically comprised of two files (sets of trained parameters for the **neural network** and code that runs the neural network). It is important to note the high computational effort (cost and time) required to train the parameters at this stage because, in a more general sense, we are performing a lossy compression of the internet.

After training the model, we have what is called the **Base Model**. At this stage, the neural network can predict the next word in a sequence or the next letter. The Base Model we will consider in this project is called the **Transformer (Generative Pre-trained Transformer)**

**Transformer:** The transformer neural network architecture will be considered in the next chapter, but briefly, we understand the mathematical computations happening in this NN and how to optimize it (i.e., adjusting the parameters iteratively to make the network better at the next word prediction task), but we do not know how these parameters (billions) collaborate to achieve this.

In the next stage, to obtain a more useful neural network, we refine the Transformer (Base Model) by fine-tuning it, i.e., replacing the dataset with labeling instructions (e.g., quality questions and answers) to train the model. A lot of evaluations are done, and labeling instructions are added again. Fine-tuning is cheaper and can be done more often.

The performance of LLMs improves with the number of parameters and the amount of text we train.

It is important to note that LLMs simply think like a cache, and an advancement is to get them to think more slowly but accurately by introducing a **reward function**. There is no reward function for LLMs except in narrow domains.

A reward function assigns a numerical value (reward) to different outputs of the model. It quantifies how desirable a particular output is.

The set of tools (calculator, browsers, code interpreters, etc.) that we use as humans are also accessible by LLMs.

**LLM Security:** We won’t delve deeply into the security of LLMs. There is a large diversity of attacks because it is very new and evolving, and over time, we should have more secure LLMs. Examples of attacks include Jailbreaking, Prompt injection, Backdoors & data poisoning, Data extraction & Privacy, etc.