Generative AI for beginners: Session I

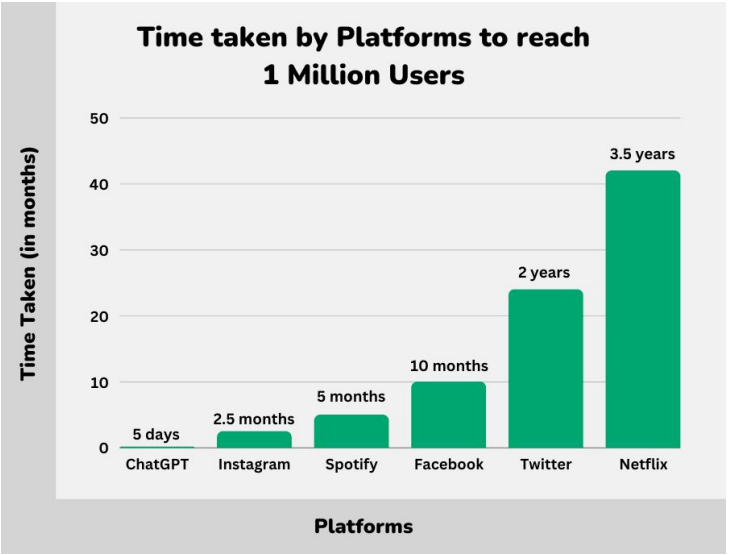
Khyati Ganatra

Senior Data Scientist at Cequence

WOMEN WHO CODE®

Have you heard about ChatGPT yet??

- Every company wants to use GenAI
  - Google, Nvidia, Microsoft, AMD, Notion, Adobe …..the list goes on and on
- We see courses, blogs, podcasts talking about using LLM models in our day-today
  - 10 Prompts everyone must know
  - Enhance productivity with ChatGPT
- Countless ways people are using these tools
- **Prompt engineer** is a job role now!!



ChatGPT made AI accessible to everyone

Agenda

- AI Landscape
- High level ML and Deep Learning concept overview
- Prompt engineering (Part 2)
- What is Generative AI?
- Generative AI in action
- Focus on LLMs:
  - Applications of LLMs
  - Peek under the hood of LLMs
  - Concerns with LLMs

AI Landscape

AI application areas:

- Robotics
- Face recognition
- Recommendation systems
- Language Translation

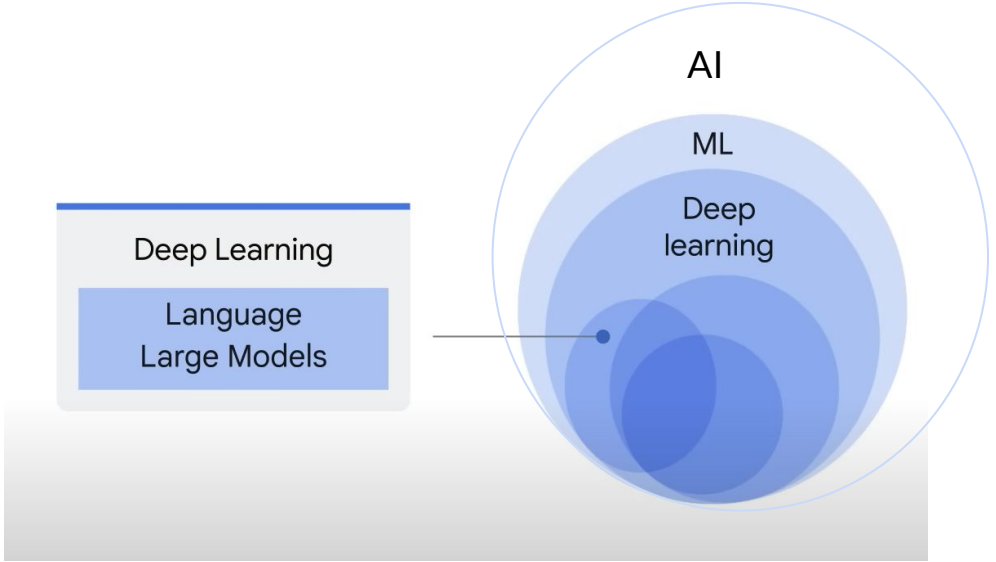


Image source: Into to Generative AI from Google skill boost

AI != ML

Artificial Intelligence is a discipline. It deals with intelligent agents that are systems can reason, and act like human.

Machine Learning is a subfield of AI

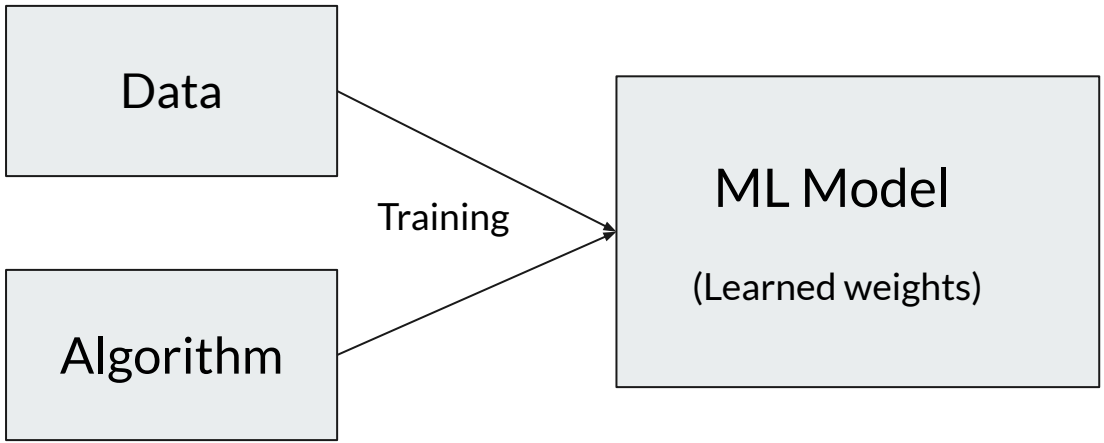
It enables a system to learn autonomously from input data (model). The trained model can then be used to make useful predictions.

Deep Learning is a type of machine learning that uses **Artificial Neural Networks** allowing them to process more complex patterns than traditional machine learning techniques

ML Approaches

	Supervised Learning	Unsupervised Learning	Reinforcement Learning
Approach	Given historic data (text/numeric), learn to make predictions for unknown (future) data	From the given data, uncover patterns and/or group similar objects together	Given a reward function, learn the actions to take to get maximum rewards
Goal	Predict Label	Find patterns in the data	Maximize reward
Labeled Data	Yes	No	No
Example	Predict house price, spam filtering	Market Basket Detection, anomaly detection	Play a game, self driving cars

ML Approaches



Deep Learning

Neural Network: Interconnected nodes (neurons) that process data as it passes through them

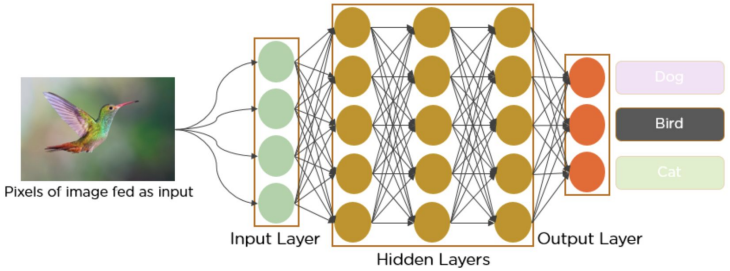
Layers: Vertical stack of filters. Each layer extracts higher level features from the data

Weights and Biases: Each connection between neurons has a weight and a bias, which determine the strength of the signal passing through the connection.

Training: The process of feeding data into the neural network, learn the characteristics of the object in the image and accurately predict the object

Loss Function: A metric that quantifies how well the neural network's predictions match the actual data. The goal is to minimize this loss during training.

Epochs: One complete pass through the entire training dataset. Multiple epochs improve the model's accuracy as it refines its understanding of the data.



Core idea behind Deep Learning is to mimic human brain in order to learn

Deep corresponds to the large number of layers between input and output

Example of an Artificial Neural Network (ANN)

Picture: Introduction to CNN (Analytics Vidya)

AI - Traditional AI vs Generative AI

Discriminative

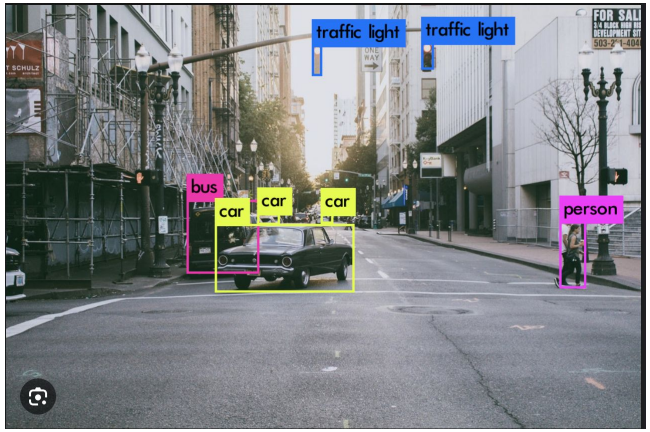
Input

Output

Dog

Avocado

Classification



Other tasks:
Sentiment analysis,
Object segmentation

AI - Traditional AI vs Generative AI

Generative

Summarize this review for me in 35 words: I bought this for my wife around the same time I bought a Pixel watch for myself. The reason? I have a Pixel phone and she has a Samsung phone. Quite frankly I found the Pixel watch easier to set up, and it has some advantages like better access to Fitbit. Also the notifications are better on the Pixel watch and I think the watch face options are better on the Pixel. But either are fine choices, and this Samsung is generally significantly less expensive which will probably be the deciding factor for most people (that or their phone).

Both use a proprietary magnetic charging disc which attaches to the phone to charge. They seem very similar, but I haven't checked to see if they are interchangeable. Much better design than some proprietary designs I've seen on some fitness trackers. Battery life is rather similar (the tech press gives the Pixel a bad knock in that regard), but probably slightly better on the Samsung. Finally, the Pixel watch comes with two sizes of watch band, which can be useful if your wrist is either smaller or larger than average.

View other drafts

Sure, here is the summary of the review in 35 words:

- **Samsung Galaxy Watch Active 2 vs. Pixel Watch:** Both are great smartwatches, but the Samsung is less expensive and has a slightly better battery life.

Google it

Input: Furniture element made from apple

Output

Text to image generation

Generated by Dall-e2

Generative AI in action!!!

Demo

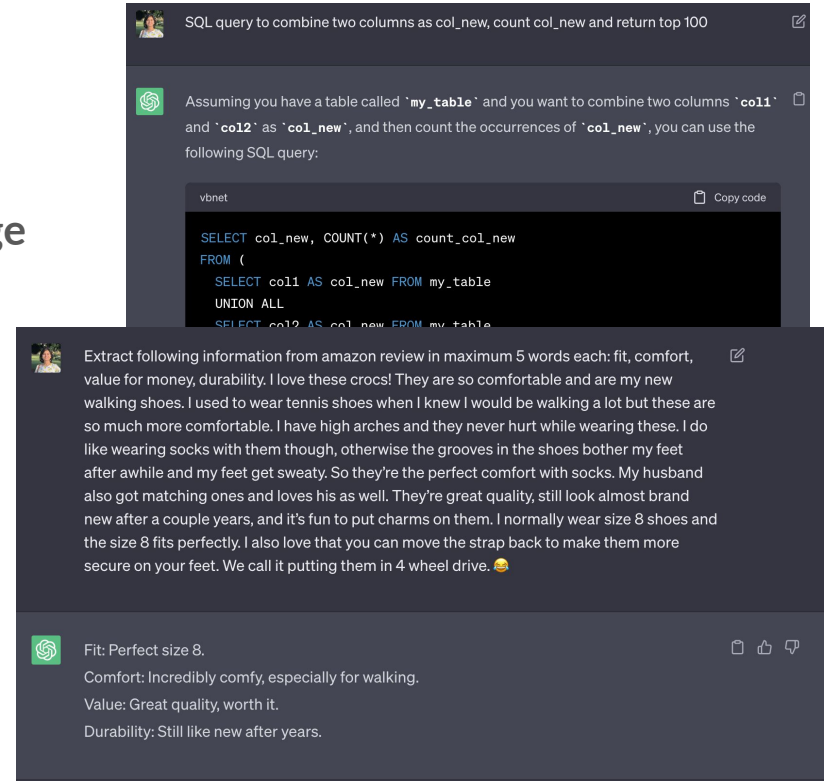
Let’s look into LLMs (Large Language Models)

Large Language Models a.k.a LLMs

- AI models that are capable of generating human like text based on the training on enormous amount of text data
  - Eg ChatGPT was trained on **547 GB** of text data
- **General purpose** LLMs.
- **A single LLM can be used for variety of tasks unlike previous approaches where we train a model for single task**
  - Eg chatbot, content creation
- **Super Heavy models** with **billions** of parameters
  - Eg Meta release Llama-2 that has **70B parameters.** Requires **100+ GB** memory to store the model

Applications of LLMs

- Most exciting application - **Natural Language Generation**
  - Predicting the next word
- Text summarization
- Translation
- Sentiment analysis
- Format transformation
- Code generation/debugging
- Novel text generation
  - eg. write a poem in Shakespeare style about San Francisco (try it out!!)



How does it do all of this???

Building blocks of LLM

Tokenization:

Convert input sentence in smaller chunks (words, subwords) that model can process

Embedding:

Convert words/tokens into numerical vector keeping the semantic meaning intact

Attention:

Helps the model to understand importance of different words in the input and thus what to focus on

Pretraining: ✖

- Training an LLM on a large dataset
- Model learns general language patterns, relationship between words, understanding language

Transfer Learning ✔

- Technique to leverage the learnings from pretraining and apply on a new but related task
- Fine-tuning LLMs for a specific task requires very less labeled data to get good performance on new task

Transfer Learning

Zero-shot Learning

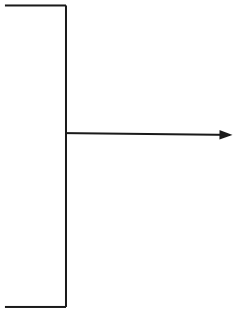
- Using the model (inference) as is

Few shot Learning

- Giving couple of examples for task at

Fine tuning

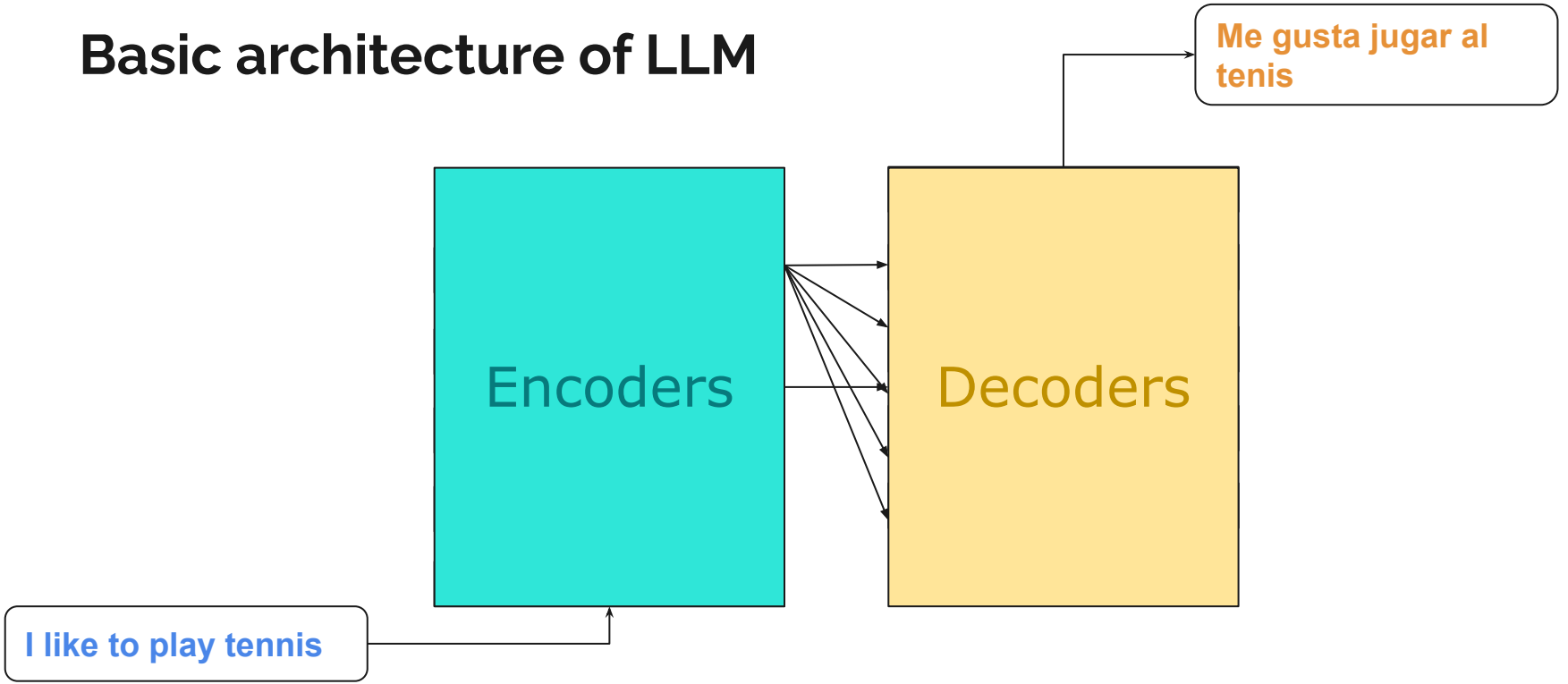
- Modifying the model weights to train it for the desired task



Prompt Engineering

(Deep dive in next session)

Basic architecture of LLM



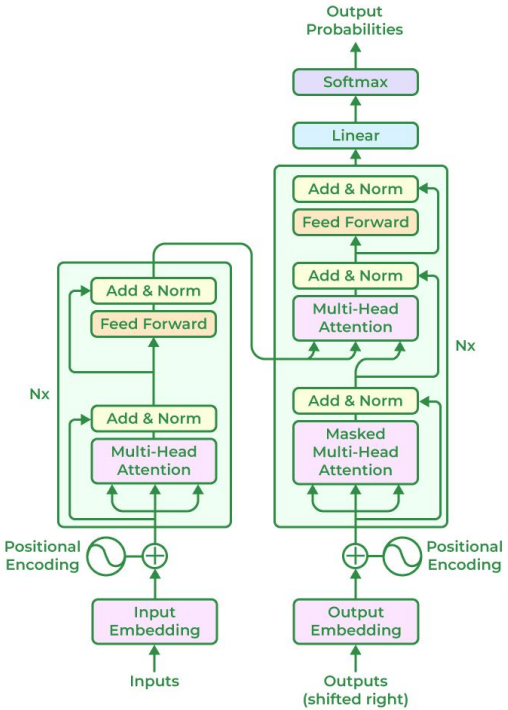
Transformers

Transformers work by first breaking the **input text** into **tokens**.

The **encoder** takes the tokens as input and creates a **vector representation of the input text**. This vector contains information about the meaning of the text, as well as the relationships between the words.

The **decoder** then takes the representation from the encoder as input and generates the output text. The decoder works by **predicting the next token in the output text, one token at a time.**

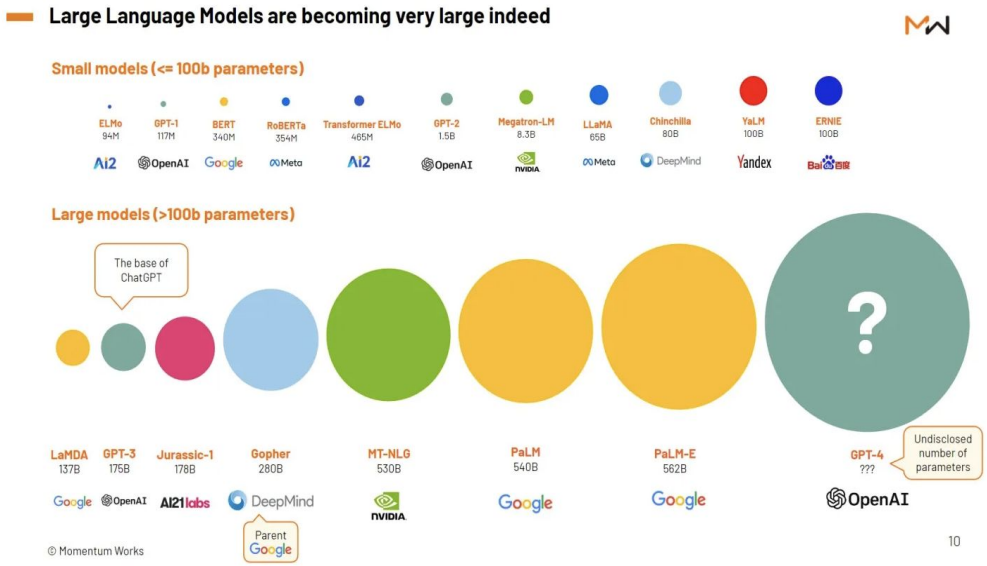
Self-attention mechanism in the transformers network allows it to focus on the important words in the input text, regardless of their position in the sentence.



Different types of LLMs

- **Autoregressive Language Models (e.g., GPT)**: Autoregressive models generate text by **predicting the next word** in a sequence given the previous words. They are trained to maximize the likelihood of each word in the training dataset, given its context. The most well-known example of an autoregressive language model is OpenAI's GPT (Generative Pre-trained Transformer) series, with GPT-4 being the latest and most powerful iteration.
- **Autoencoding Language Models (e.g., BERT)**: Autoencoding models, on the other hand, learn to **generate a fixed-size vector representation (also called embeddings) of input** text by reconstructing the original input from a masked or corrupted version of it. They are trained to predict missing or masked words in the input text by leveraging the surrounding context. BERT (Bidirectional Encoder Representations from Transformers), developed by Google, is one of the most famous autoencoding language models. It can be fine-tuned for a variety of NLP tasks, such as sentiment analysis, named entity recognition, and question answering.

Most popular/widely used LLMs



Using these models:

- OpenAI provides APIs to integrate/use ChatGPT in different applications
- Hugging face has a hub of open-source pre-trained models. It provides a python library transformers that enables us to use these models via API calls
- We can further fine-tune these models or directly use for inference

Concerns with LLMs

Training related:

- Training $$$ due to heavy compute and large data requirement
- Let's not forget the carbon-footprint of these ever growing models
- Catastrophic forgetting - Fine-tuning for specific tasks makes it forget some other general purpose tasks
- Hallucinations - makes up stuff on its own

Output related:

- Models are stochastic (probabilistic) in nature. So output can vary at different times
- Generate toxic content. Making the model output more helpful and less harmful is an active area of research
- Can be used for malicious intent

Resources for further reading

Books: Generative Deep Learning (2nd edition) by David Foster

Free (for limited time) courses:

Introduction to Generativa AI (Coursera - Beginner)

Generative AI with LLMs (Coursera - Intermediate)

Reference articles/blogs:

- https://vitalflux.com/large-language-models-concepts-examples/#How_does_LLM_work_Key_Building_Blocks
- https://jalammar.github.io/visualizing-neural-machine-translation-mechanics-of-seq2seq-models-with-attention/
- https://towardsdatascience.com/transformers-explained-visually-part-1-overview-of-functionality-95a6dd460452

Thank you!

Q&A