Transformers (Decoder Block)

Zeham Management
Technologies BootCamp by
SDAIA

September 10th, 2024

Objectives

By the end of this module, trainees will have a comprehensive understanding of:

- ✓ Understand Transformer Architecture
  Apply Transformers to NLP Tasks
- ✓ Differentiate Between Recurrent-Based and Transformer Attention
- ✓ Evaluate Model Performance
- ✓ Fine-Tune Pre-trained Models
- ✓ Implement Custom Transformers
- ✓ Interpret Attention Mechanisms
- ✓ Optimize Hyperparameters

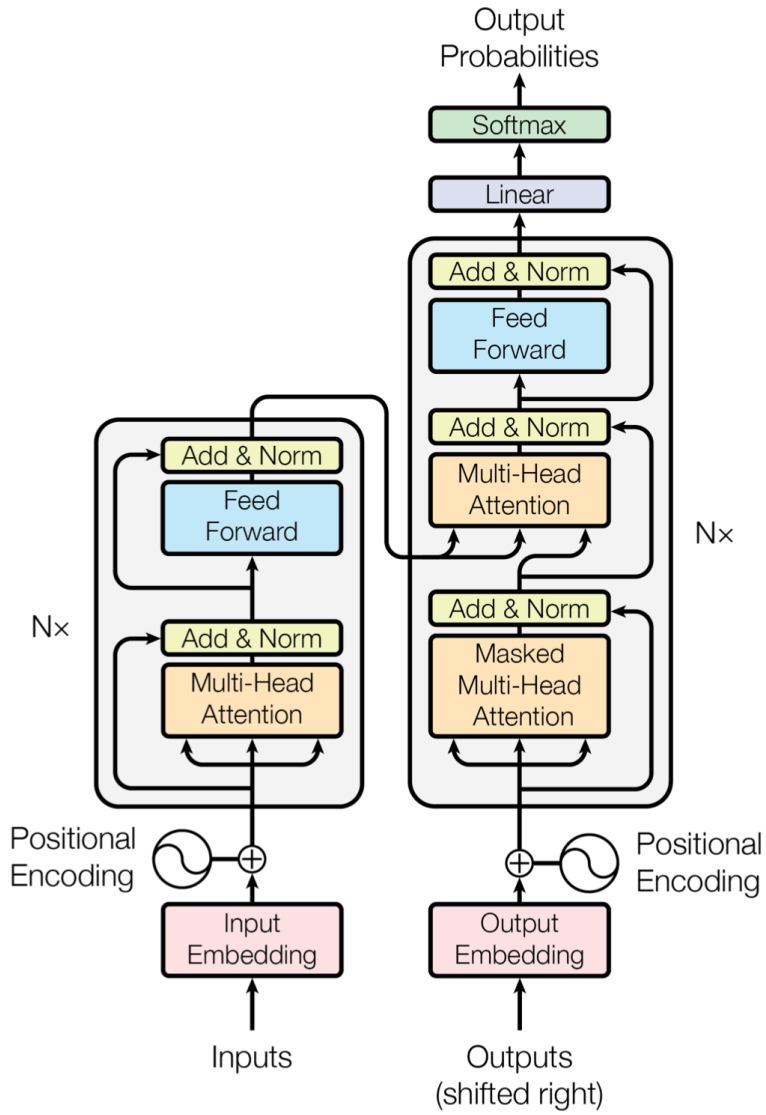
Agenda

- Decoder
- Hello Transformer!
- References

Decoder

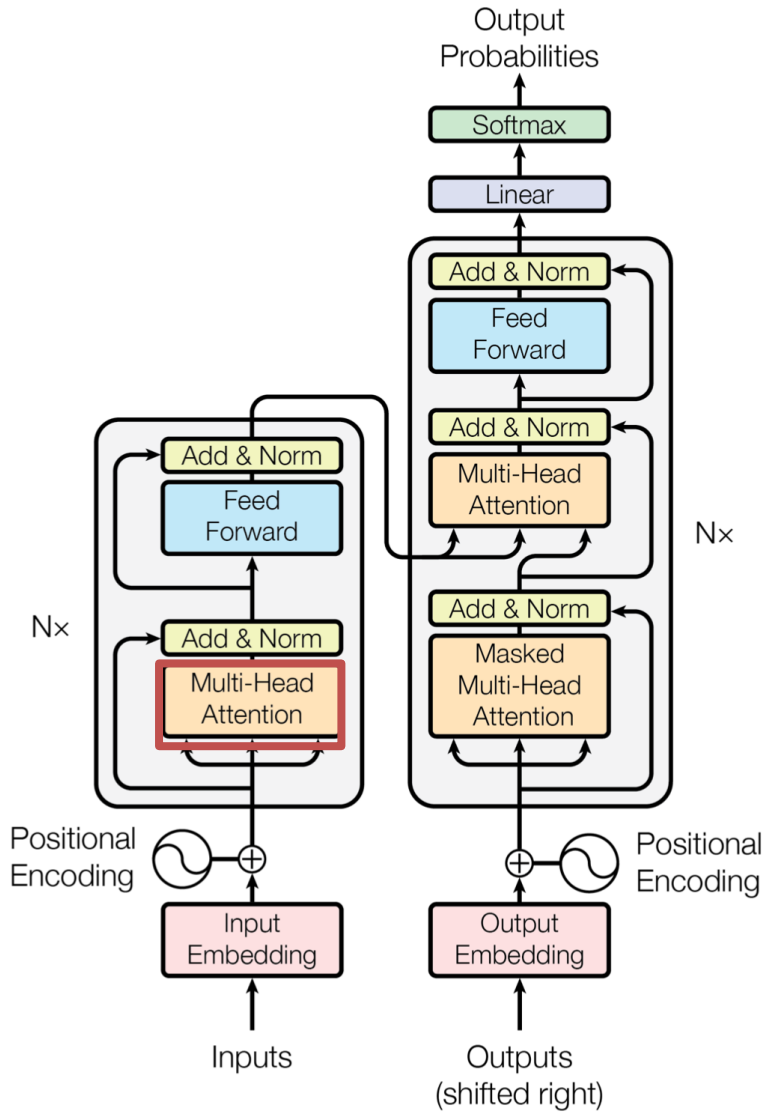
Decoder

The decoder also contains a stack of *N*=6 layers. In addition to the encoder structure, the decoder contains one more sub-layer, which is a Masked Multi-Head Attention.



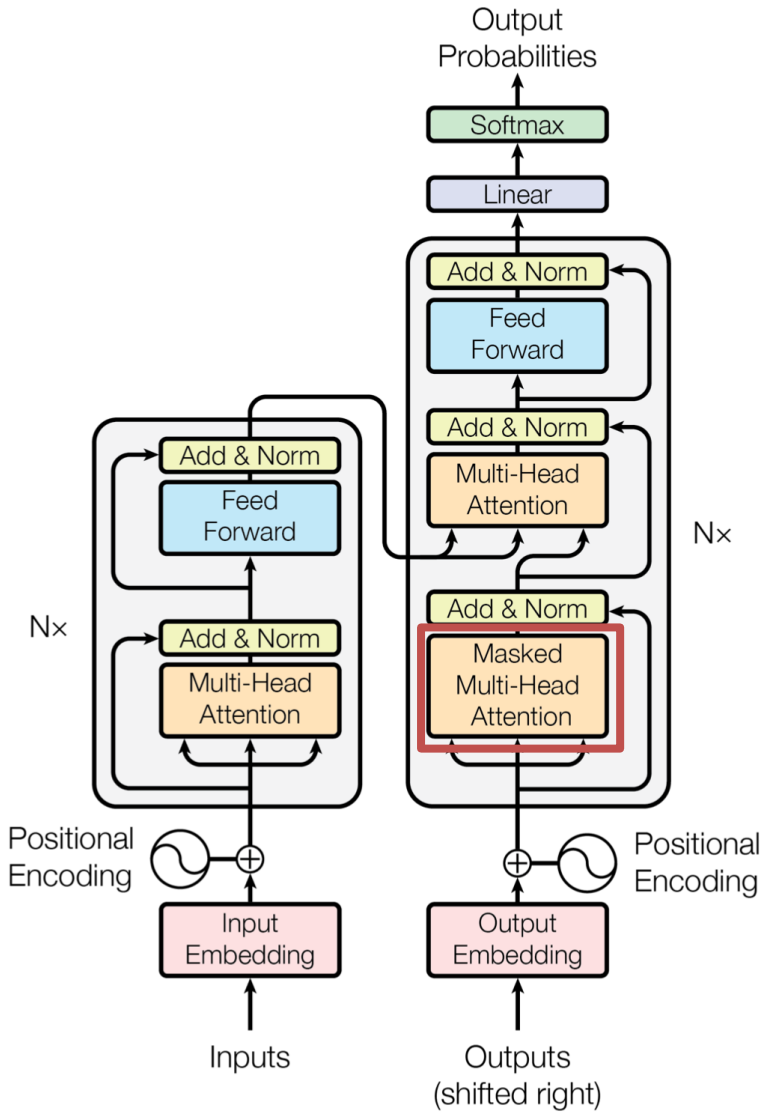
Decoder

In the encoder, the Multi-Head Attention each output can be computed by complete input sequence.



Decoder

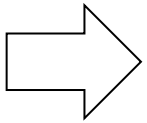
A decoder must only have access from the beginning of the sequence until and including the current position for output to be computed.



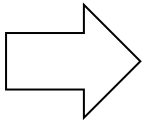
Decoder

This is achieved by masking the attention scores of the sequence that comes after the current position.

	I	love	natural	language	processing
I	1.00	0.63	0.48	0.29	0.18
love	0.63	1.00	0.72	0.41	0.25
natural	0.48	0.72	1.00	0.58	0.34
language	0.29	0.41	0.58	1.00	0.62
processing	0.18	0.25	0.34	0.62	1.00



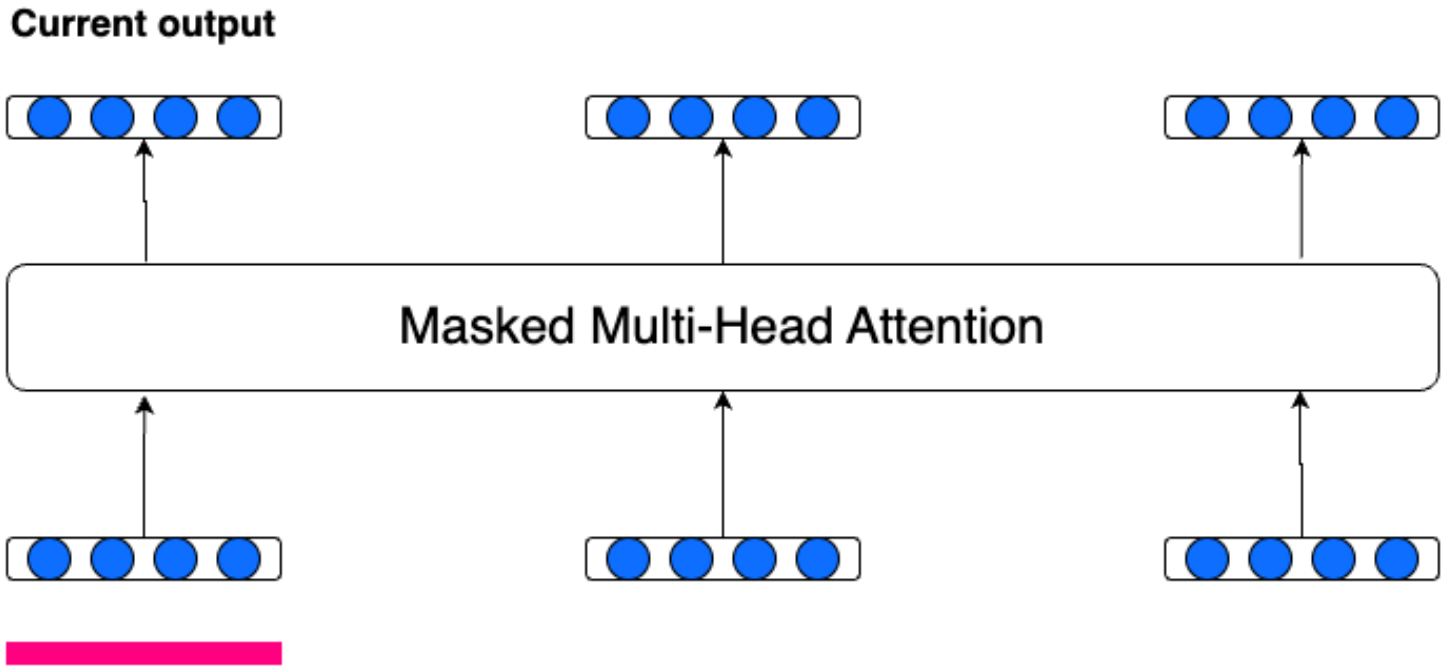
	I	love	natural	language	processing
I	1.00	0	0	0	0
love	0.63	1.00	0	0	0
natural	0.48	0.72	1.00	0	0
language	0.29	0.41	0.58	1.00	0
processing	0.18	0.25	0.34	0.62	1.00



SOFTMAX

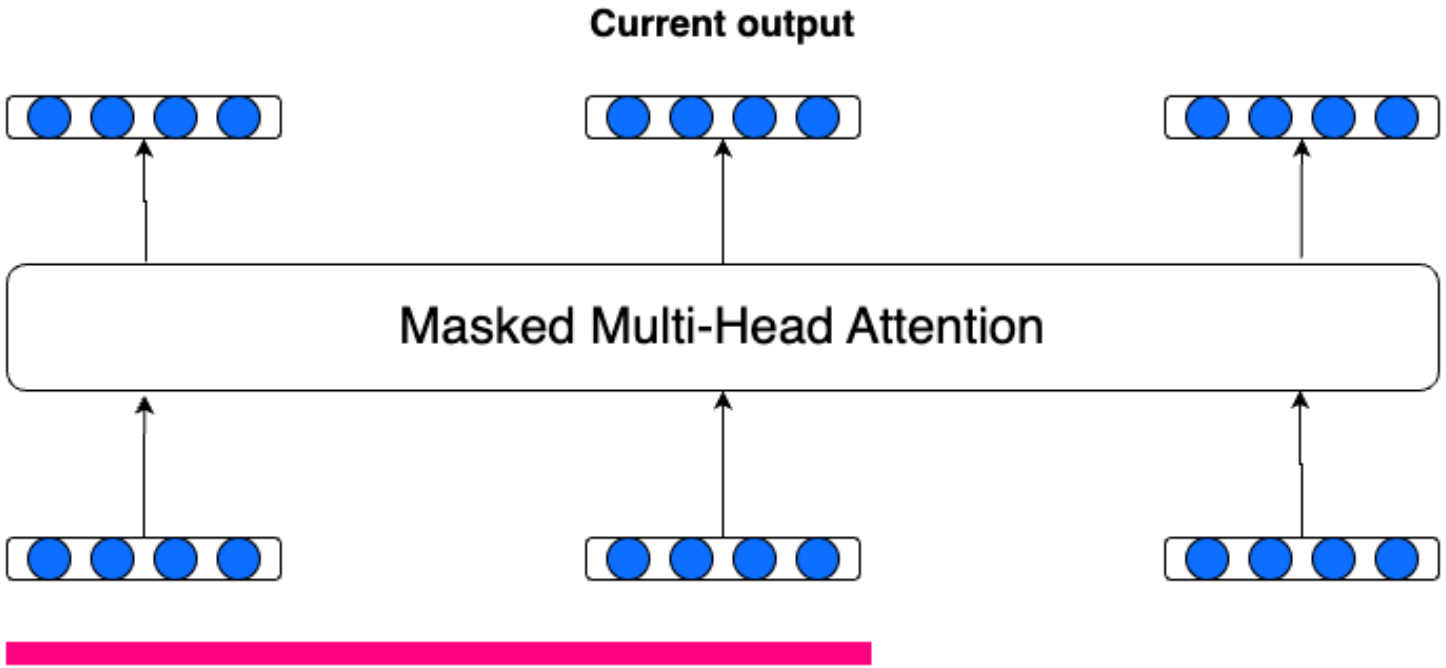
Decoder

Now the output would only consider from the beginning of the sequence until and including the current position.



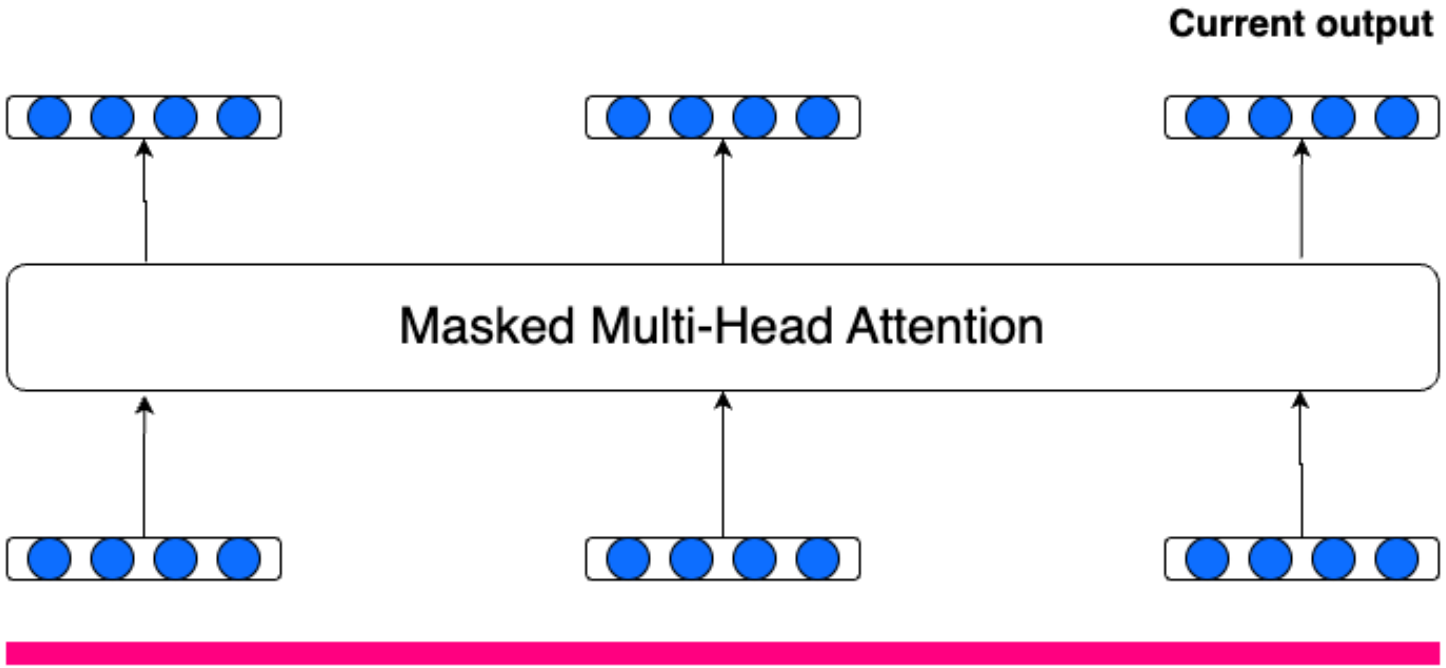
Decoder

Now the output would only consider from the beginning of the sequence until and including the current position.



Decoder

Now the output would only consider from the beginning of the sequence until and including the current position.



Large Language Models

What is Large Language Model?

GPT, BERT, and T5 (The Transformer) are Large Language Models. A Large Language Model is an advanced type of artificial intelligence that is trained on large amounts of text data to understand, generate, and interact using human language.

What is Large Language Model?

LLM properties:

- Large scale with billions of parameters.
- Use deep learning techniques.
- Pre-trained on diverse datasets and fine-tuned for specific tasks.

What is Large Language Model?

These LLMs are trained on large datasets. One gigabyte of text can contain approximately 178 million words. A large language model typically trains on datasets that are one petabyte or lager, which is about one million times larger than one gigabyte.

What is Large Language Model?

Parameters in the context of LLMs are numerical values that are learned during the training process. They are the weights and biases in the model's Neural Network that adjusts as the model learns more from the data.

What is Large Language Model?

A famous example to LLMs would be OpenAI’s chatGPT-3 which is trained on approximately 45 Terabytes of data, with 175 billion parameters

Large Language Models Components

A large Language Model contains three main components, which are:

1. Data
2. Architecture
3. Training

Large Language Models Components (data)

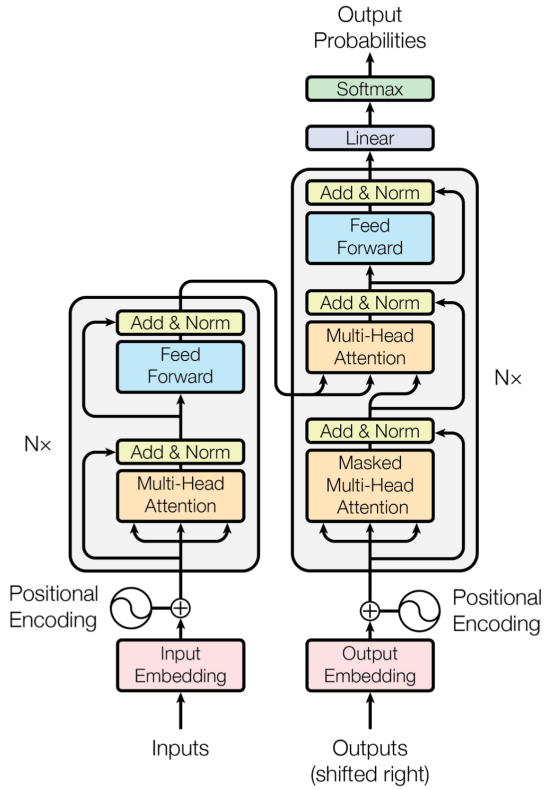
The data typically consist of a large corpus that are sourced from multiple places such as:

- Scraped websites.
- Wikipedia articles.
- News archives.
- Code repositories.

Large Language Models Components (architecture)

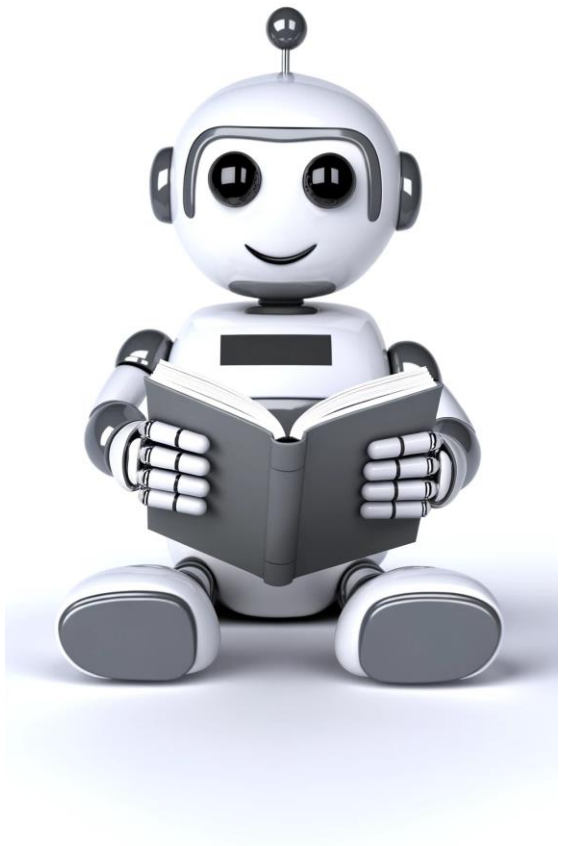
The architecture of LLMs typically relies on the Transformer architecture, which enables the model to handle sequences of data.

Transformers are specifically designed to understand the context of each word in a sentence by considering its surrounding words.



Large Language Models Components (Training)

In the training process, this architecture iteratively learns from the gathered large dataset, adjusting the model's parameters to improve its predictions to the actual output.



LLM vs NLP

Differences between LLM and NLP:

- LLM is a subset of NLP, focusing on specific models within the broader field of language processing.

- LLM emphasizes text generation and comprehension, whereas NLP encompasses a wider array of tasks such as text analysis, language recognition, and translation.

LLM vs NLP

Differences between LLM and NLP:

- LLM models typically exhibit greater complexity, necessitating larger datasets and computational resources compared to conventional NLP systems.

- LLM excels in tasks requiring deep linguistic understanding and text generation, while NLP finds application across various domains including text mining, sentiment analysis, and chatbot development.

Hello Transformer!

`Building_A_Transformer_From_Scratch_Decoder.ipynb`