# Transformers

## Agenda

1. Use Cases
2. High-Level Architecture
3. Tokenization
4. Attention Mechanisms
5. Position Embeddings
6. Project 1 - Sentiment Analysis with Transformer from Scratch via PyTorch

7. Popular Variants - BERT and RoBERTa
8. Project 2 - Text Summarization with BERT Models from HuggingFace
9. Knowledge Distillation
10. Popular Variants - DistilBERT
11. Increasing Context Windows - RoPE and Flash Attention
12. Fine-tuning
13. Efficient Fine-tuning with Low Rank Adaptation (LoRa)
14. Popular Variants - T5
15. Project 3 - Fine-tuning T5 for Named Entity Recognition (NER) with Autotrain

16. Generalized Pretrained Transformer (GPT)
17. Project 4 - Building GPT from Scratch with PyTorch and Lightning AI
18. Alignment - RLHF and DPO
19. Project 5 - Improving GPT Responses with DPO
20. Large Language Models (LLMs)

21. Popular Variants - Llama Architecture
22. Project 6 - Fine-tuning Llama 3.3 8B for Medical Question-Answering with LitGPT

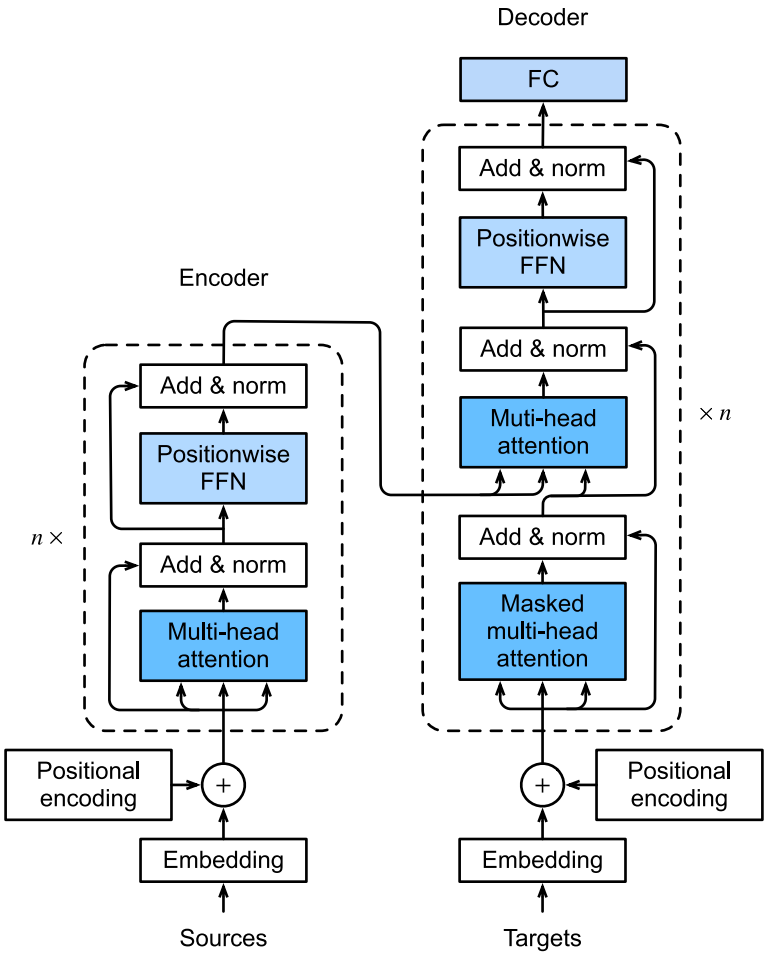
23. Popular Variants - DeepSeek
24. Alignment Variant - GRPO
25. Project 7 - Conducting Local Inference with DeepSeek via Ollama

# Use Cases

1. Machine Translation
    - Translating spoken / written languages.
    - Converting one programming language to another.
2. Question Answering
    - Answering questions based on a given context.

3. Text Generation
    - Generating text based on a given prompt.
    - Summarizing long texts.
4. Classification
    - Classifying text into categories.
    - Sentiment analysis.
5. Named Entity Recognition
    - Identifying and classifying entities in text.

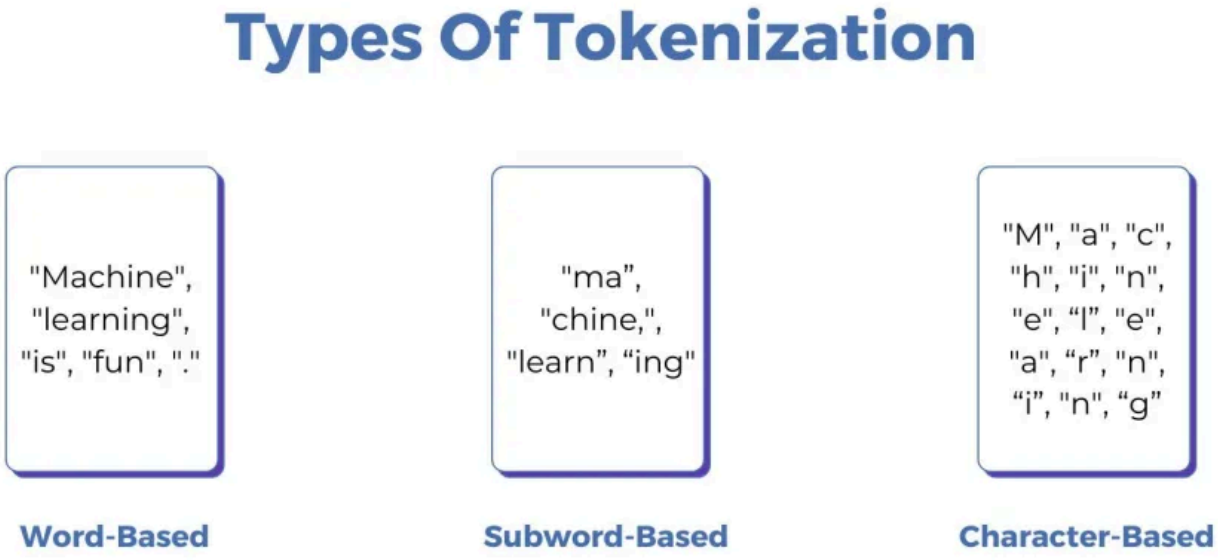
# Transformer Architecture




1. Encoder
    - **Input**: Sequence of tokens from your data.
    - **Output**: Sequence of embeddings to provide context to the decoder.
2. Decoder
    - **Input**: Sequence of tokens from your data.
    - **Output**: Sequence of tokens to generate text or probability score for a given task.

# Tokenization

1. Tokenization
    - Splitting text into tokens.
    - Converting tokens into embeddings.
2. What is a token?
    - A token is a word or a subword or a character in language modeling.
3. What is an embedding?
    - An embedding is a vector representation of a token in language modeling.




4. Tokenizer
    - A tokenizer is a function that converts text into tokens.
5. Popular Tokenizers

- Word Tokenizers
- Sentence Tokenizers
- Byte Pair Encoding (BPE)

# Python Examples

1. `nltk` tokenization
    - Word Tokenizer
    - Sentence Tokenizer
2. `PyTorch` implementation of `Byte Pair Encoding`

# Attention Mechanisms

1. Self Attention
    - A mechanism to allow the model to focus on different positions of the input sequence.
2. Multi-head Attention
    - A mechanism to allow the model to focus on different positions of the input sequence, but with multiple attention heads for parallelization.
3. Masked Multi-head Attention
    - Contrasts multi-head attention by introducing a mask to prevent the model from attending to future tokens.