GPT

GPT stands for "Generative Pre-trained Transformer." It's a type of artificial intelligence model developed by OpenAI that is designed to generate human-like text based on the input it receives. GPT models are built using a machine learning framework known as transformers, which are highly effective at handling sequences of data, such as natural language.

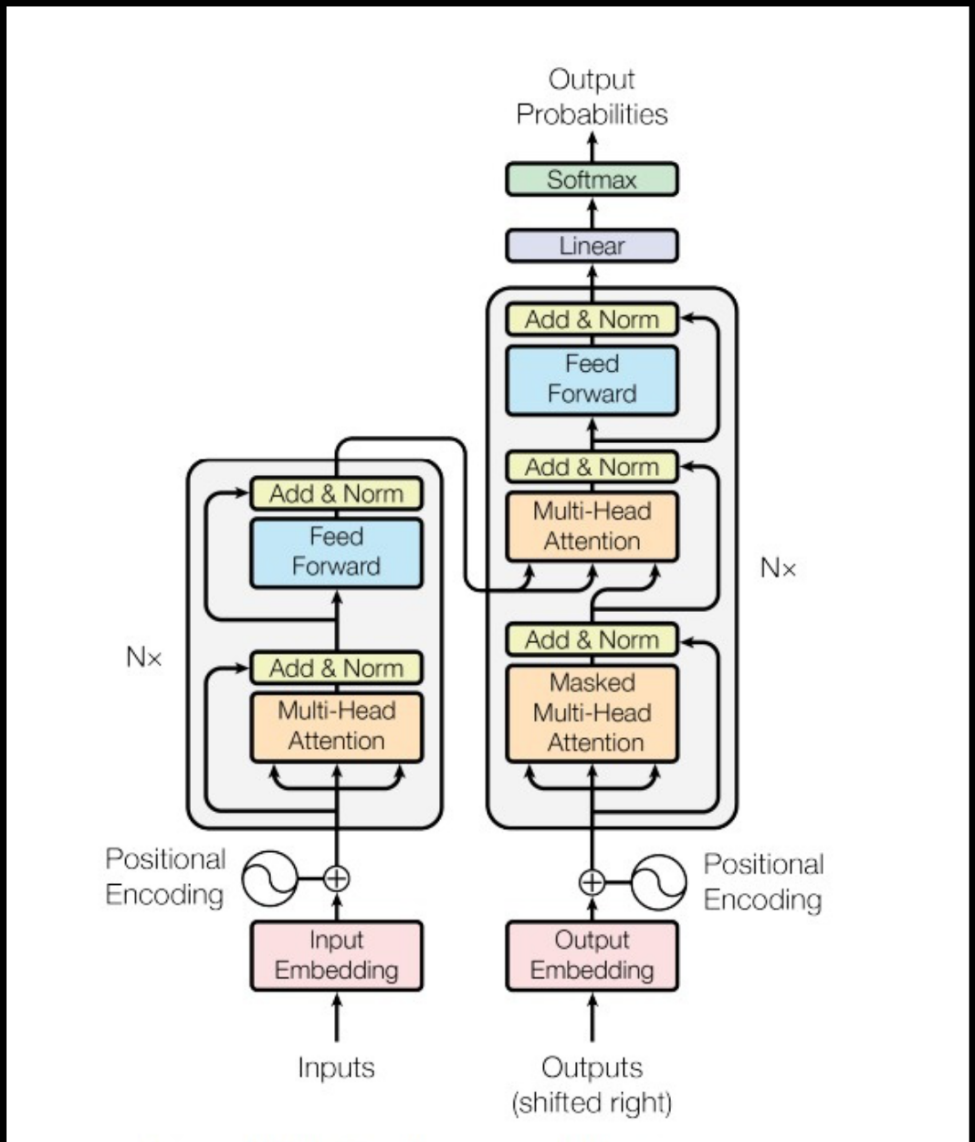
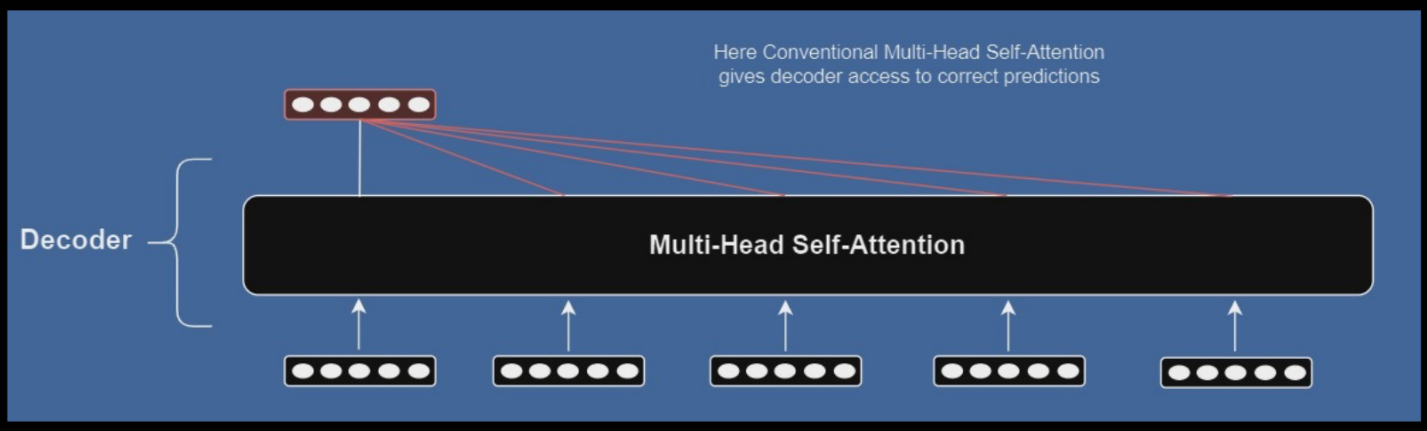
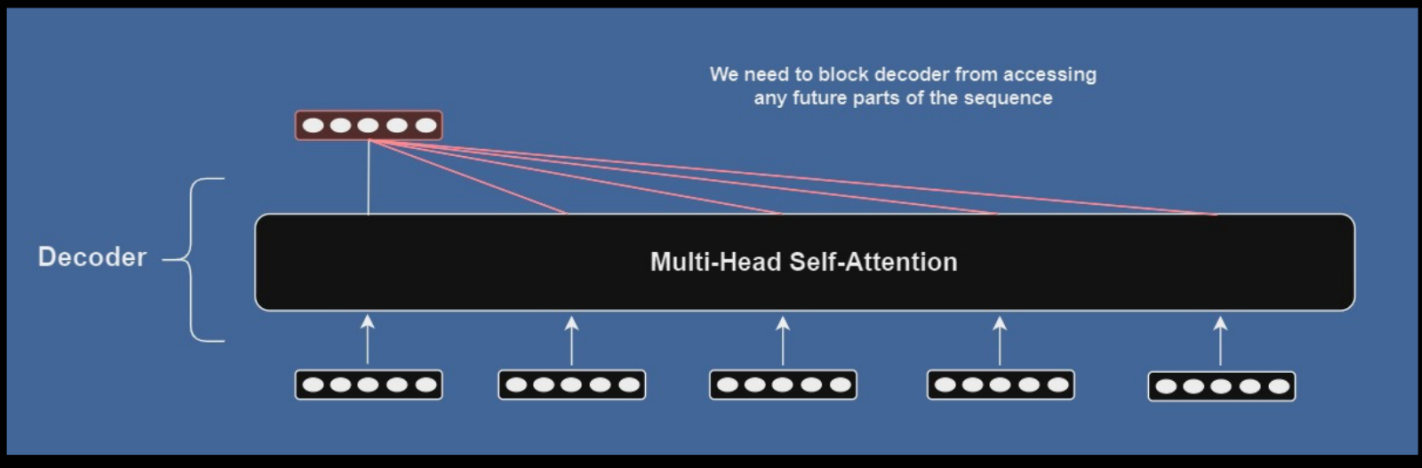
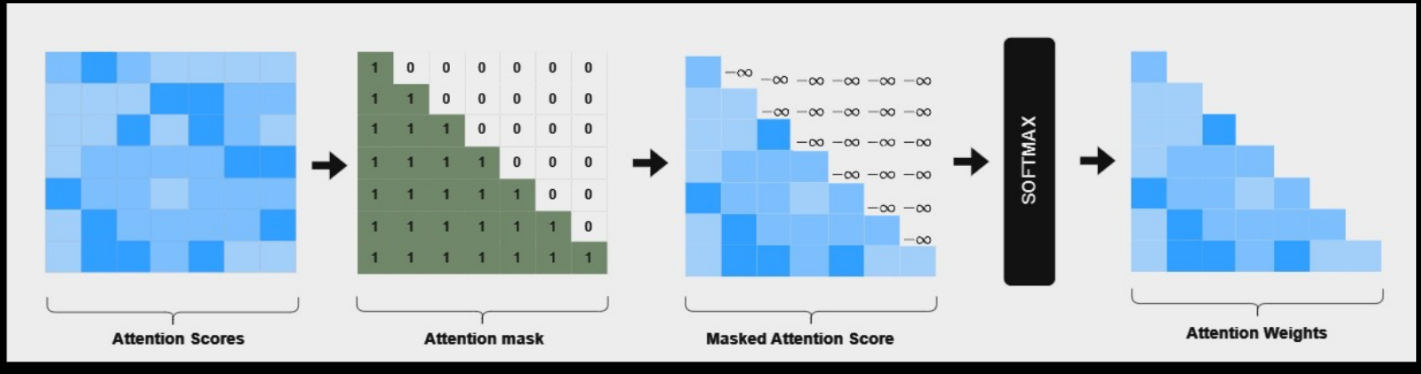
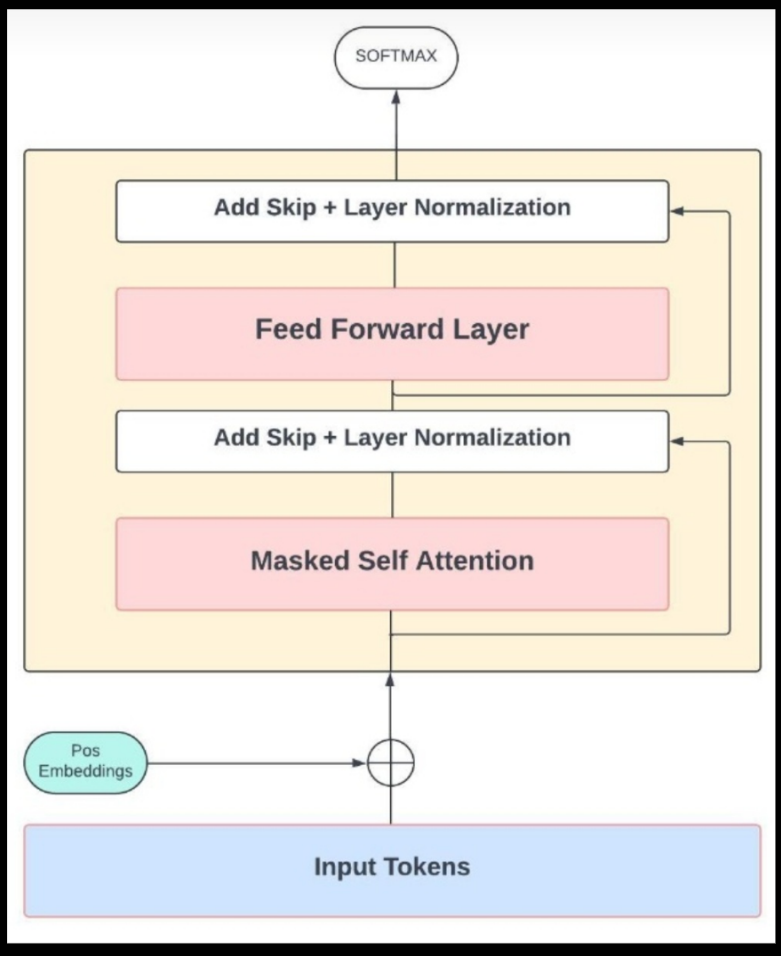


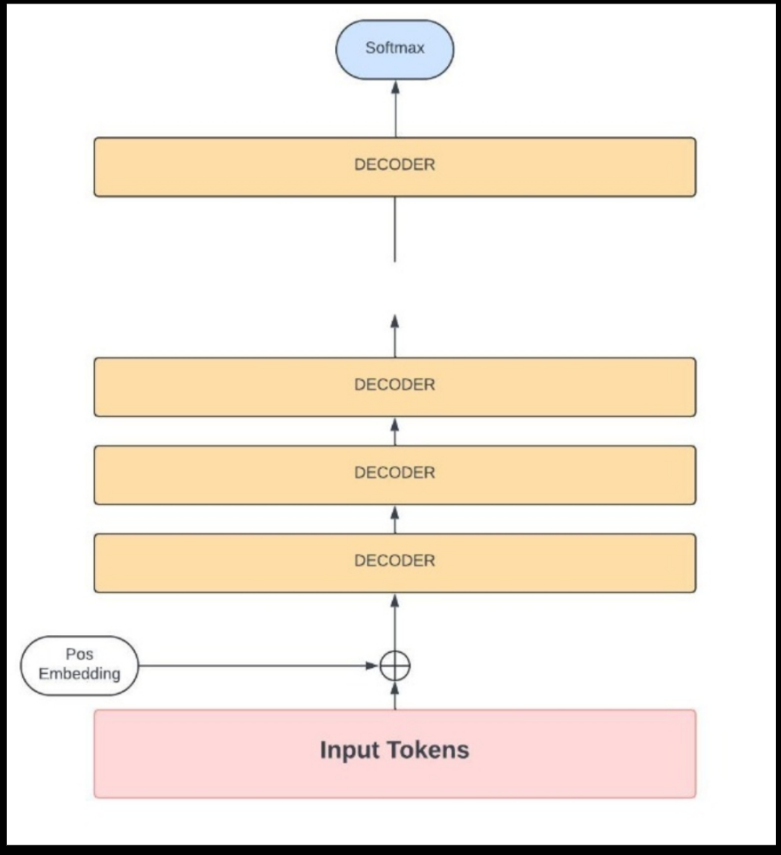
Figure 1: The Transformer - model architecture.


Here Conventional Multi-Head Self-Attention
gives decoder access to correct predictions
Decoder
Multi-Head Self-Attention


We need to block decoder from accessing
any future parts of the sequence
Decoder
Multi-Head Self-Attention


SOFTMAX
Attention Scores
Attention mask
Masked Attention Score
Attention Weights


SOFTMAX
Add Skip + Layer Normalization
Feed Forward Layer
Add Skip + Layer Normalization
Masked Self Attention
Pos Embeddings
Input Tokens


Softmax
DECODER
DECODER
DECODER
DECODER
Pos Embedding
Input Tokens

Pre-Training

1. Tokenization: Tokenize the sentence into discrete tokens based on the tokenizer's granularity ["all", "is", "well"]
2. Prepare Training Examples: Create training pairs where each input sequence precedes the target token:
Input: ["all"], Target: "is"
Input: ["all", "is"], Target: "well"
3. Apply Causal Masking: Ensure that each token in the input can only attend to previous tokens using causal (triangular) masking in the self-attention layers. This prevents tokens from attending to future tokens during prediction.

4. Model Training:
   - Process Inputs: Each input sequence is fed through the model's self-attention and feed-forward layers.
   - Predict Next Token: The model predicts the next token as a probability distribution over the vocabulary, using only the seen tokens.
   - Calculate Loss: Compare the prediction to the target token using cross-entropy loss.
   - Optimize: Backpropagate the loss and update the model's weights with an optimizer like Adam.
5. Iterate Over Diverse Data: Though this example uses a single sentence, in practice, GPT models are trained on large and varied datasets to capture complex language patterns and improve predictive accuracy, crucial for generating coherent and contextually appropriate text.

Training Phase

During training, the GPT model is typically trained using a technique called teacher forcing:

Teacher Forcing: This method involves training the model to predict the next word in a sequence using the actual previous words from the training dataset as input. The sequence is known and fixed, providing the model with the "correct" context up to the point of each prediction.

Example: For the sentence "The cat sits on the mat," the model would be trained on inputs like:

Input: "The", Target: "cat"
Input: "The cat", Target: "sits"
Input: "The cat sits", Target: "on"
And so forth...
This approach allows the model to effectively learn the probabilities of a word given the previous words in a sequence.

Inference Phase

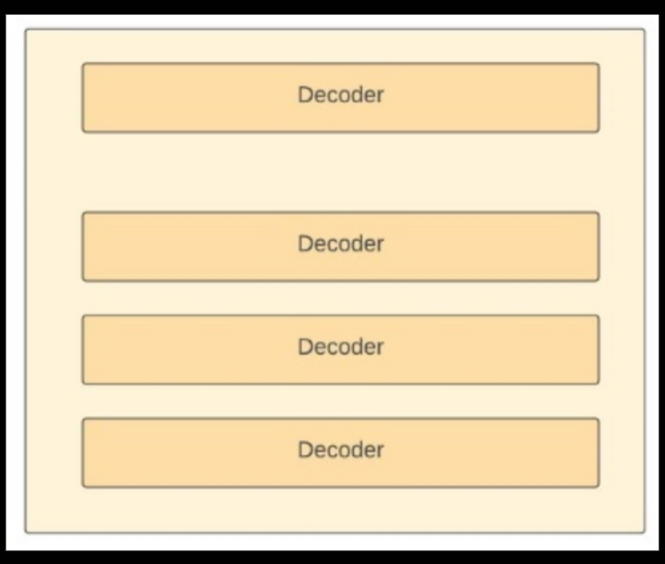
During inference, the situation changes because there is no "correct" next word provided by a training dataset. The model must generate text based on its own previous predictions:

Autoregressive Generation: The model uses its predictions as the input for generating subsequent words.

Example: If you prompt the model with "The cat", and it predicts "sits", then "sits" is fed back into the model as part of the input for predicting the next word:

Input: "The cat", Model generates: "sits"
Input: "The cat sits", Model generates: "on"
Input: "The cat sits on", Model generates: "the"
Input: "The cat sits on the", Model generates: "mat"

Fine-tuning: After pre-training, GPT models can be fine-tuned on specific datasets to perform particular tasks like translation, question-answering, summarization, etc. This



stage involves supervised learning, adjusting the model to specific requirements