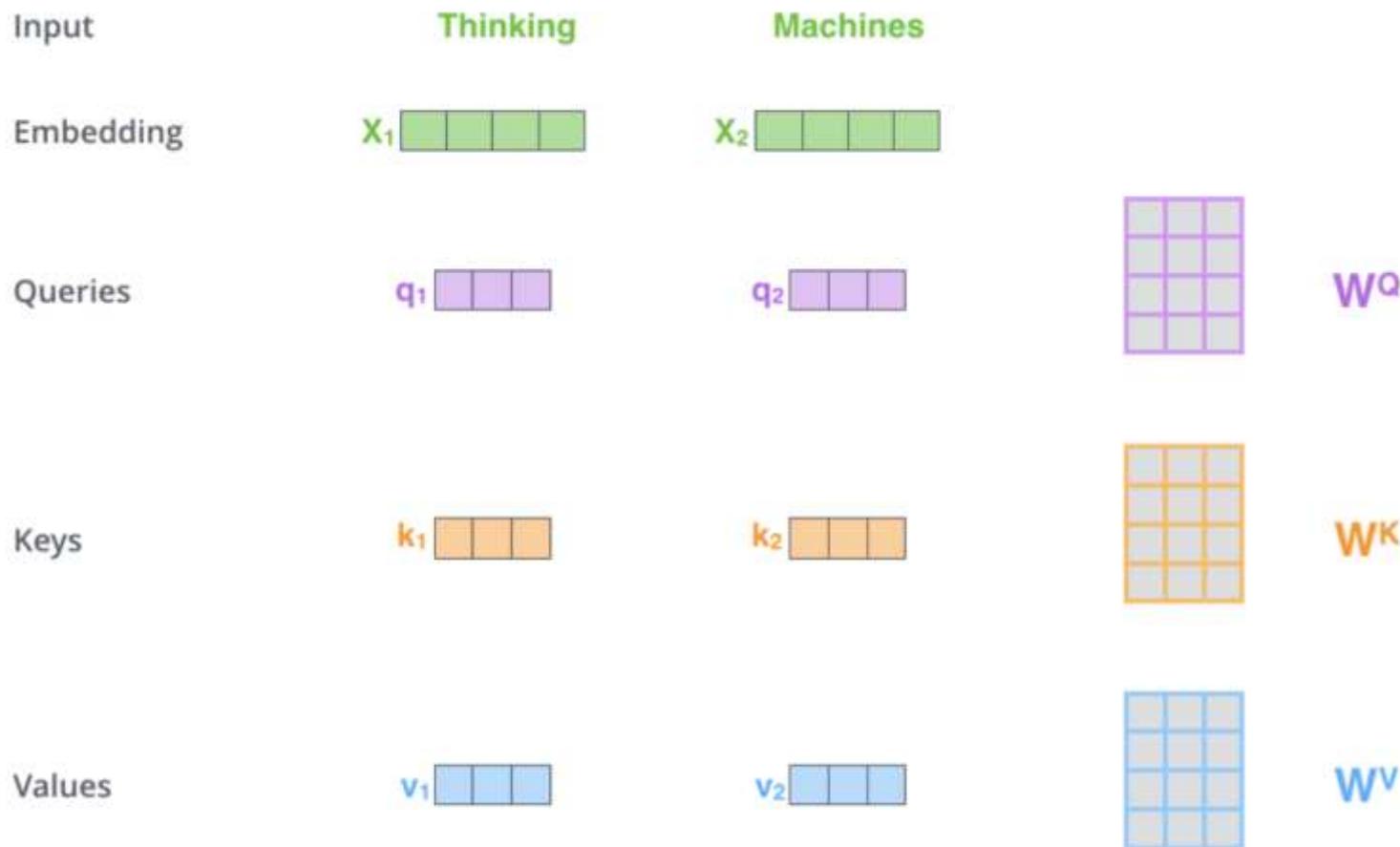


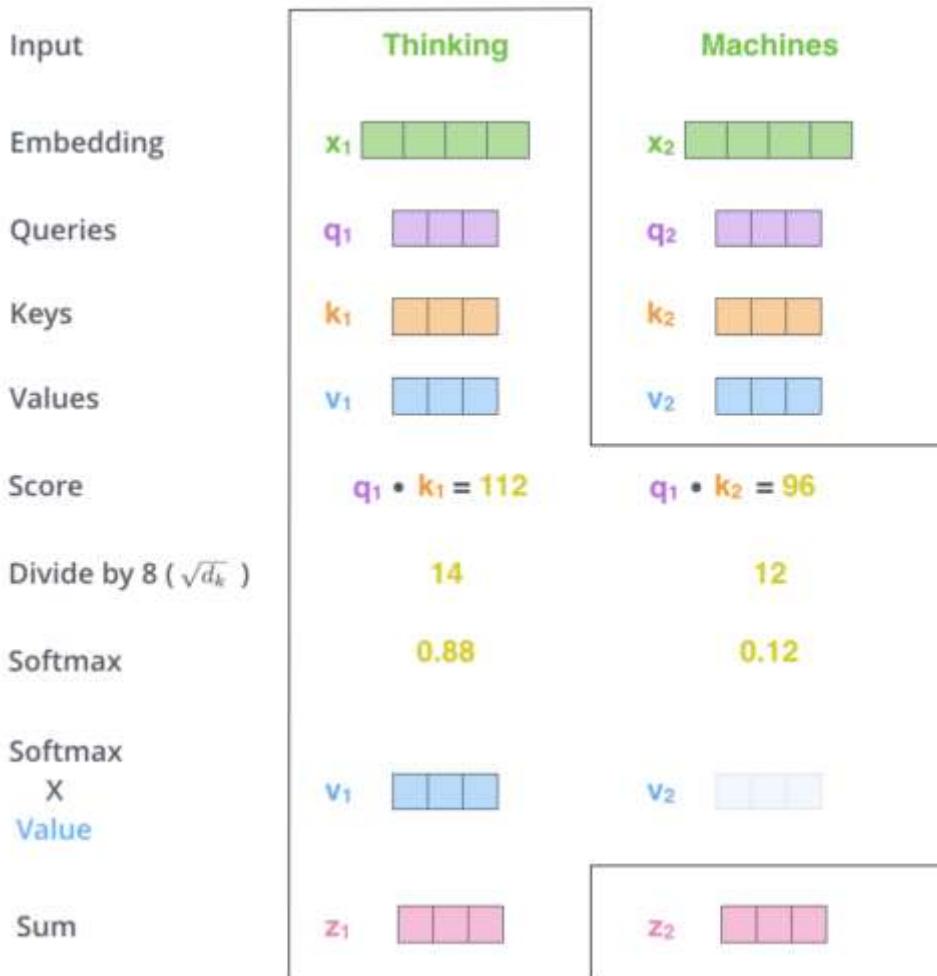
Introduction to LLMs

Attention

Attention

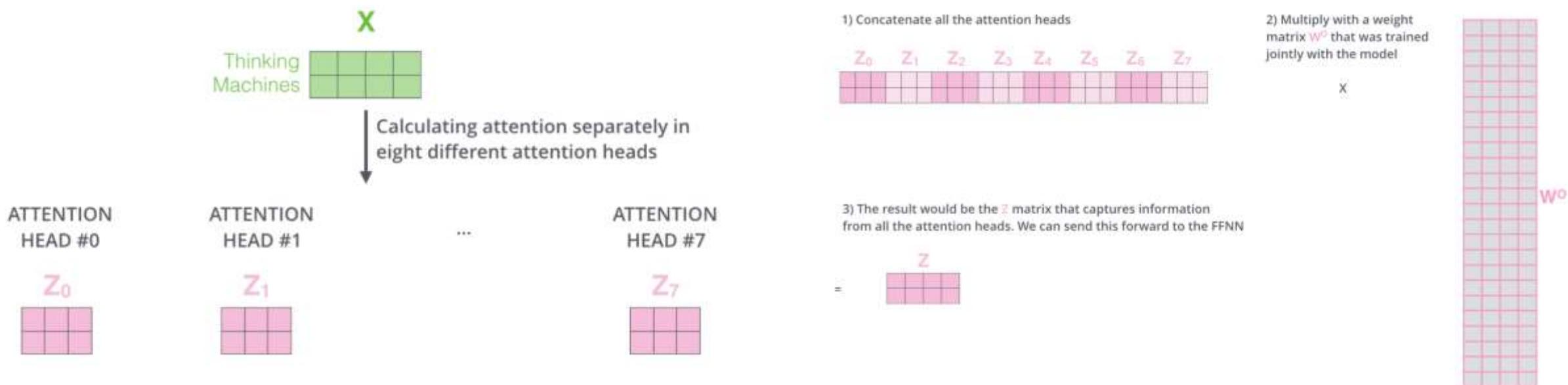


Attention

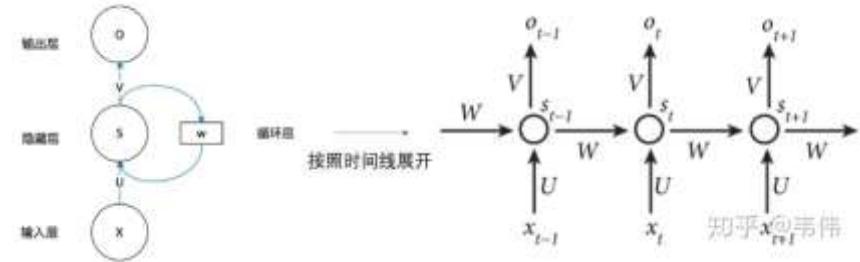


$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

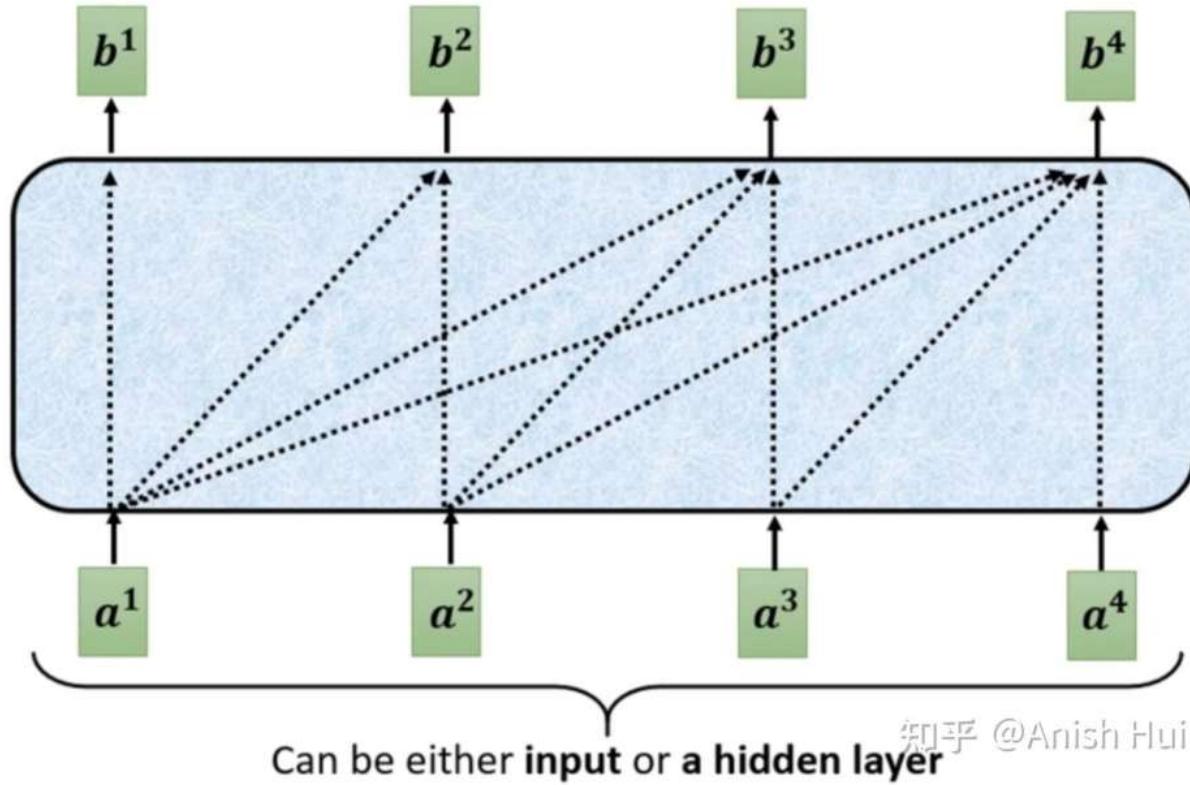
Multi-head attention



Masked self-attention



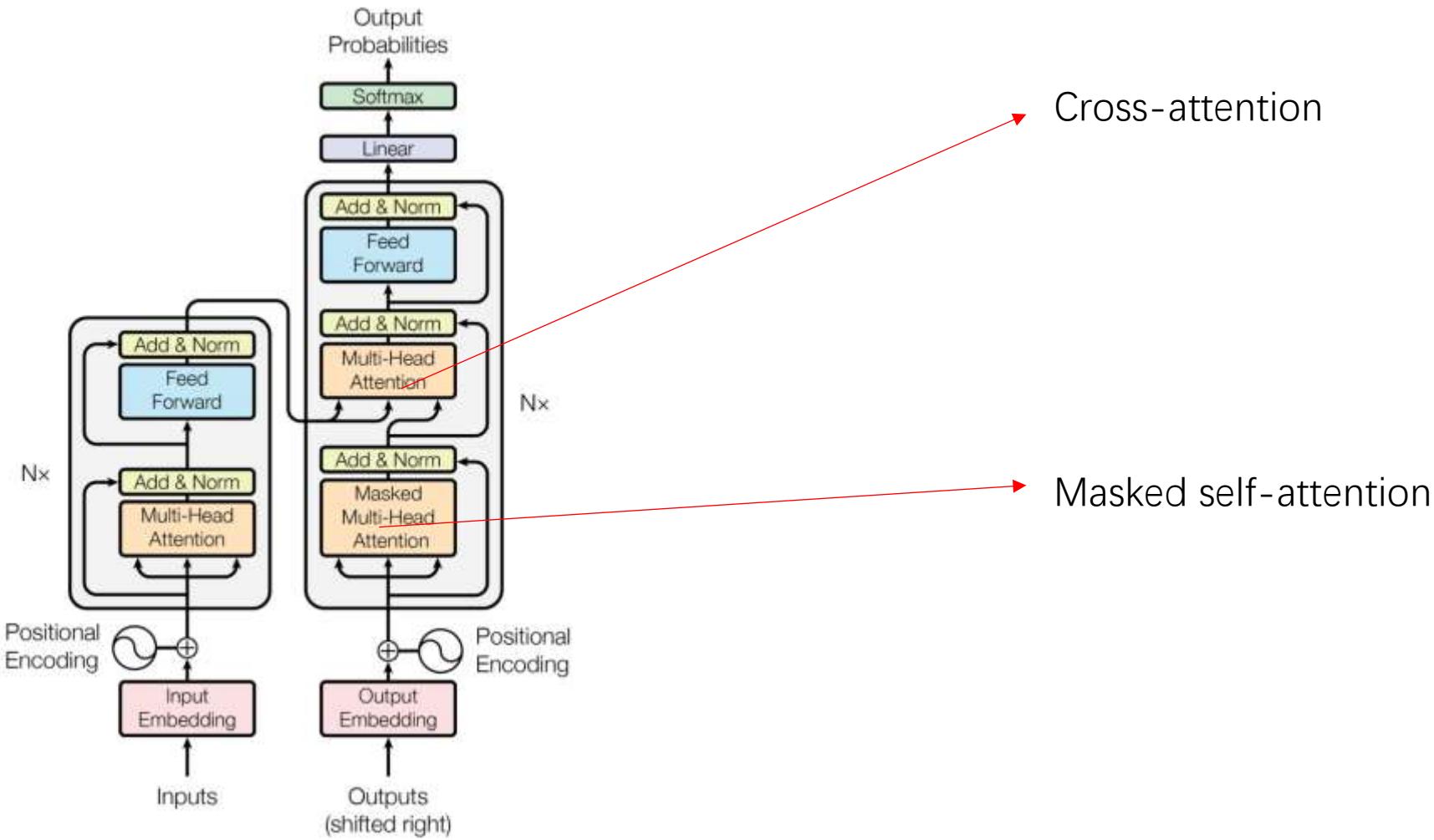
Self-attention → Masked Self-attention



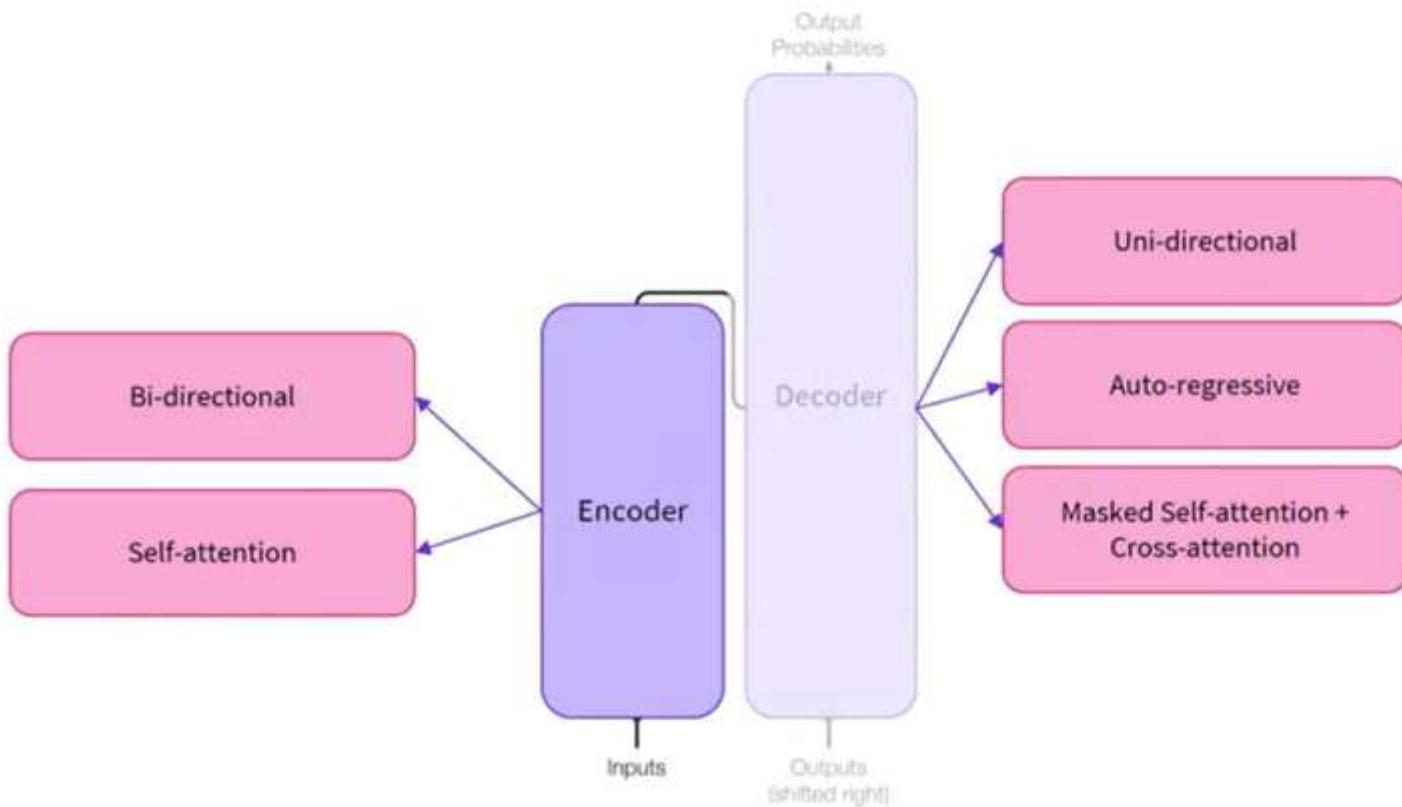
Ensures causal dependency

Transformer Models

Encoder-Decoder



Encoder-only



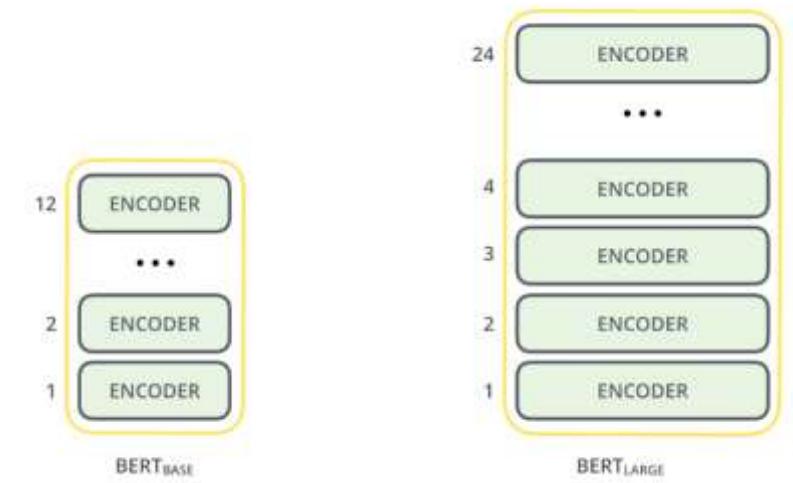
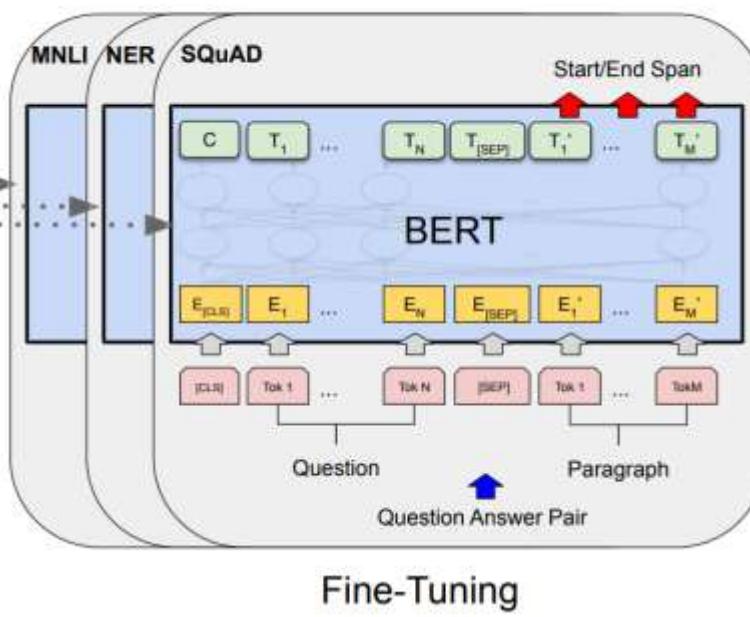
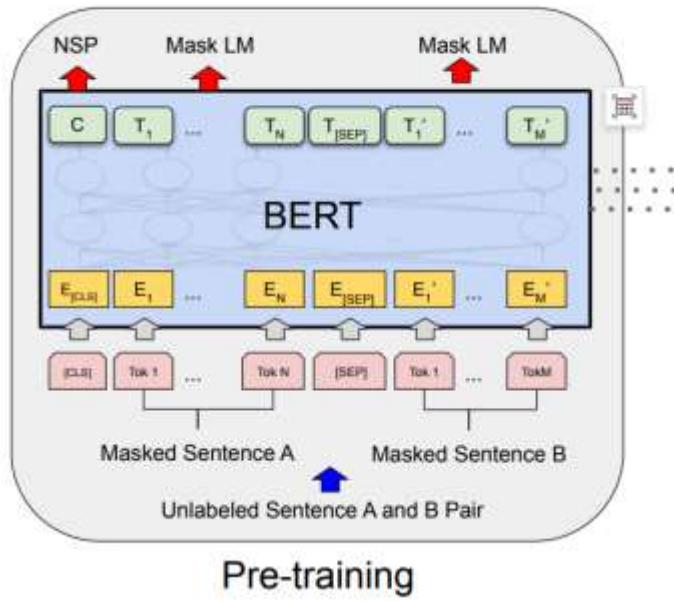
Training method:
predict masked words

Advantages:
comprehension

Disadvantages:
generation

Bert

BERT: Bidirectional Encoder Representations from Transformers

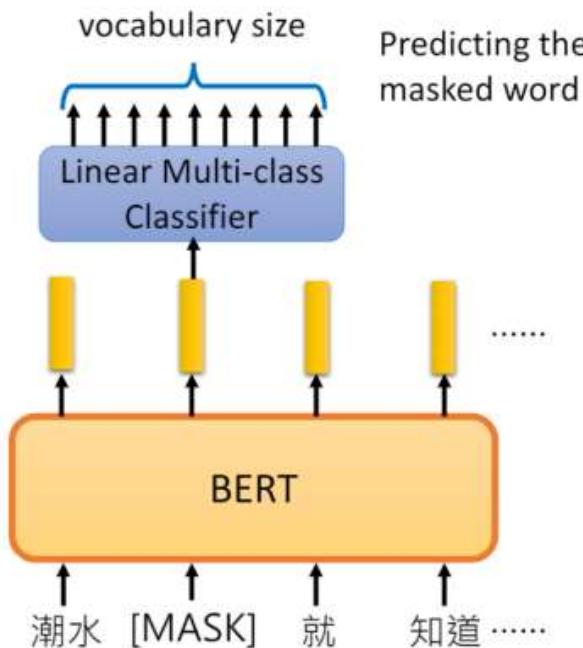


Encoder-only structure

Bert

Pretrain:

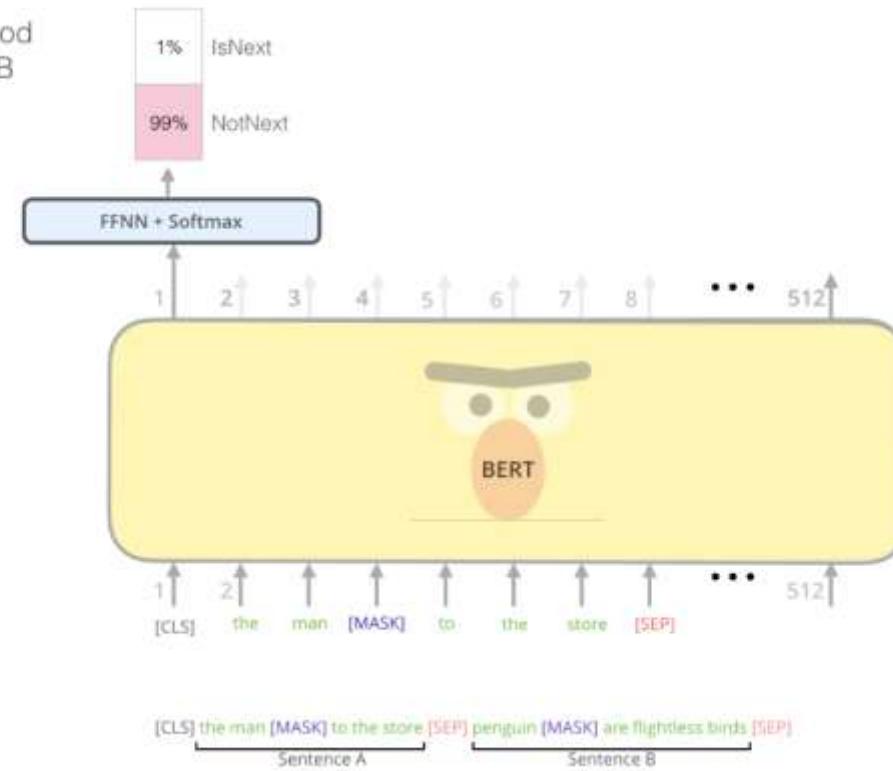
- 1、Masked LM
- 2、Next sentence prediction



Predict likelihood
that sentence B
belongs after
sentence A

Tokenized
Input

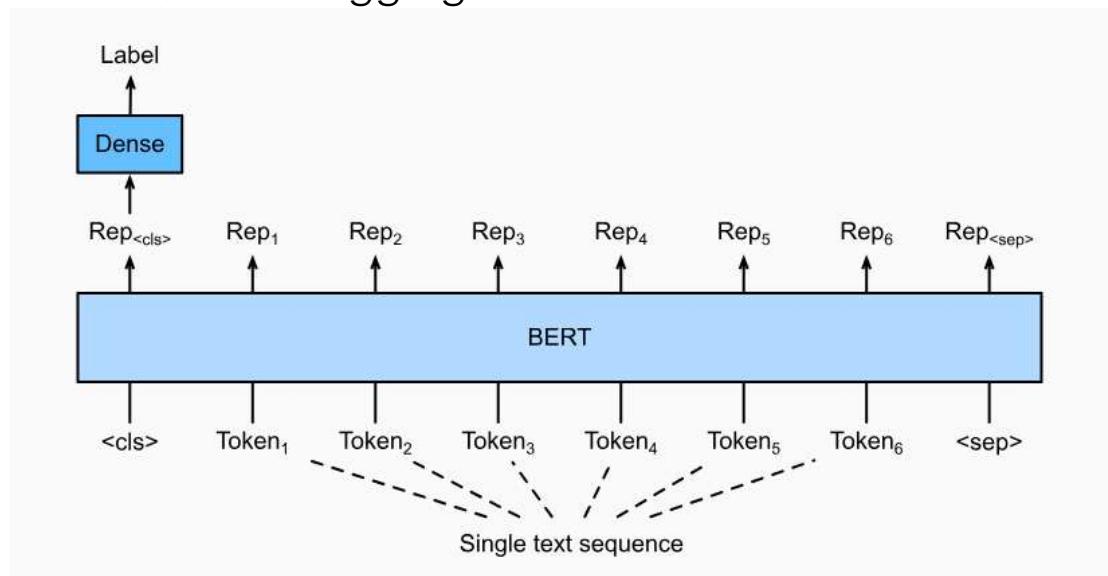
Input



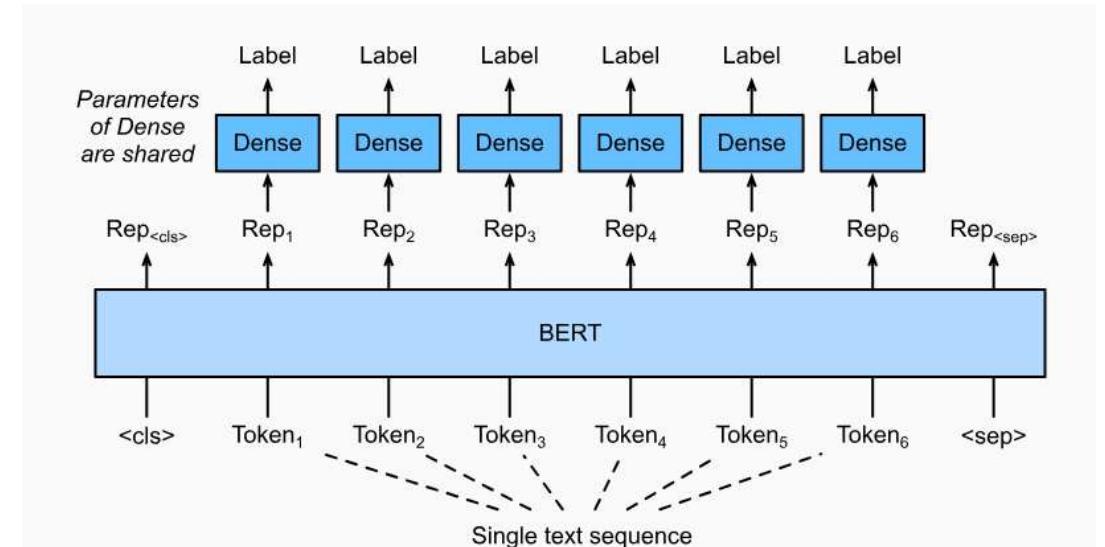
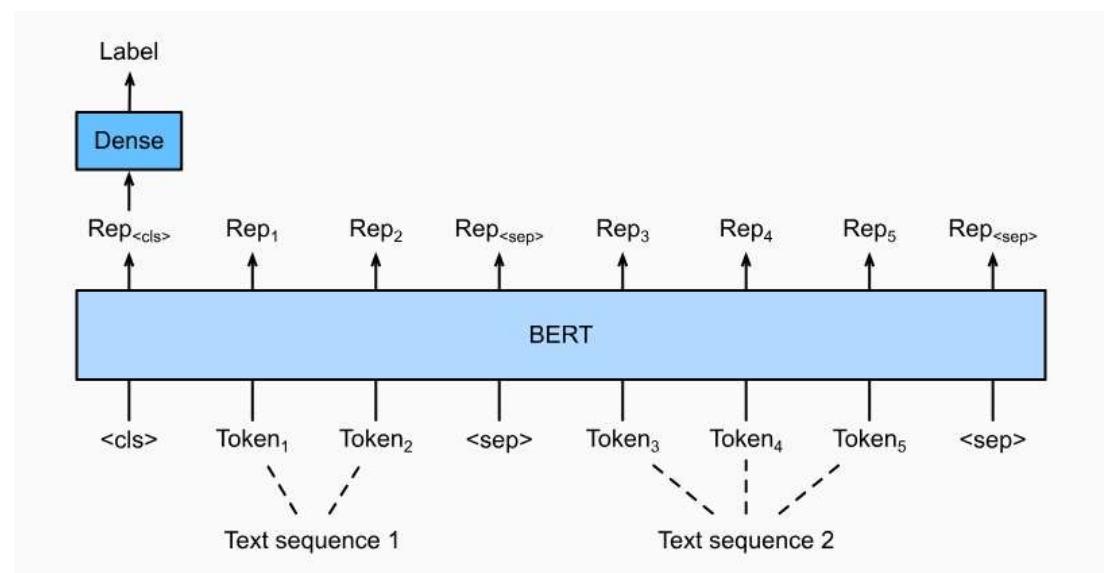
Bert

Fine-tuning:

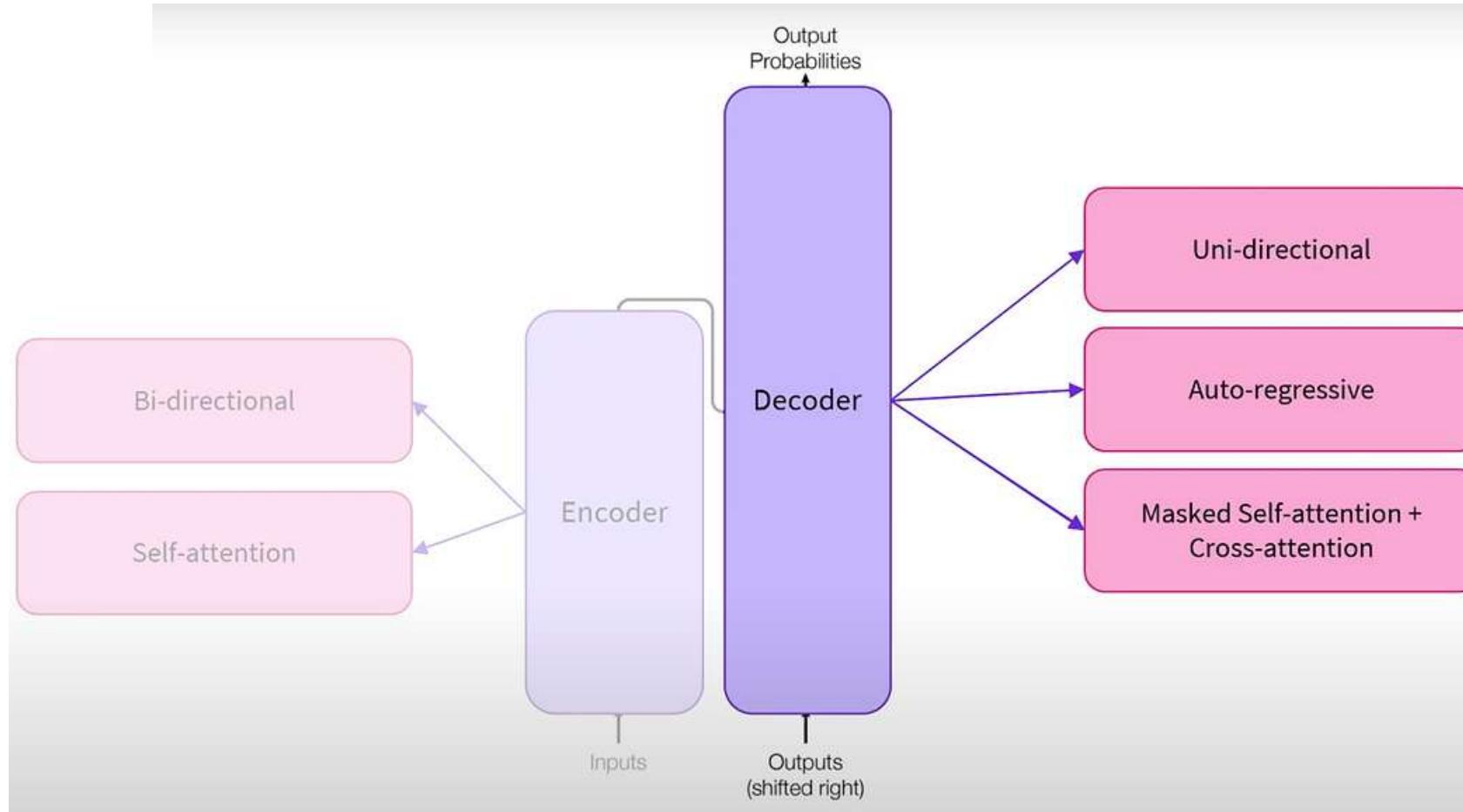
- 1、Single text classification
- 2、Text Pair Classification or Regression
- 3、Text Tagging



Highly dependent on fine-tuning
✗ Generation/Zero-shot/Few-shot



Decoder only

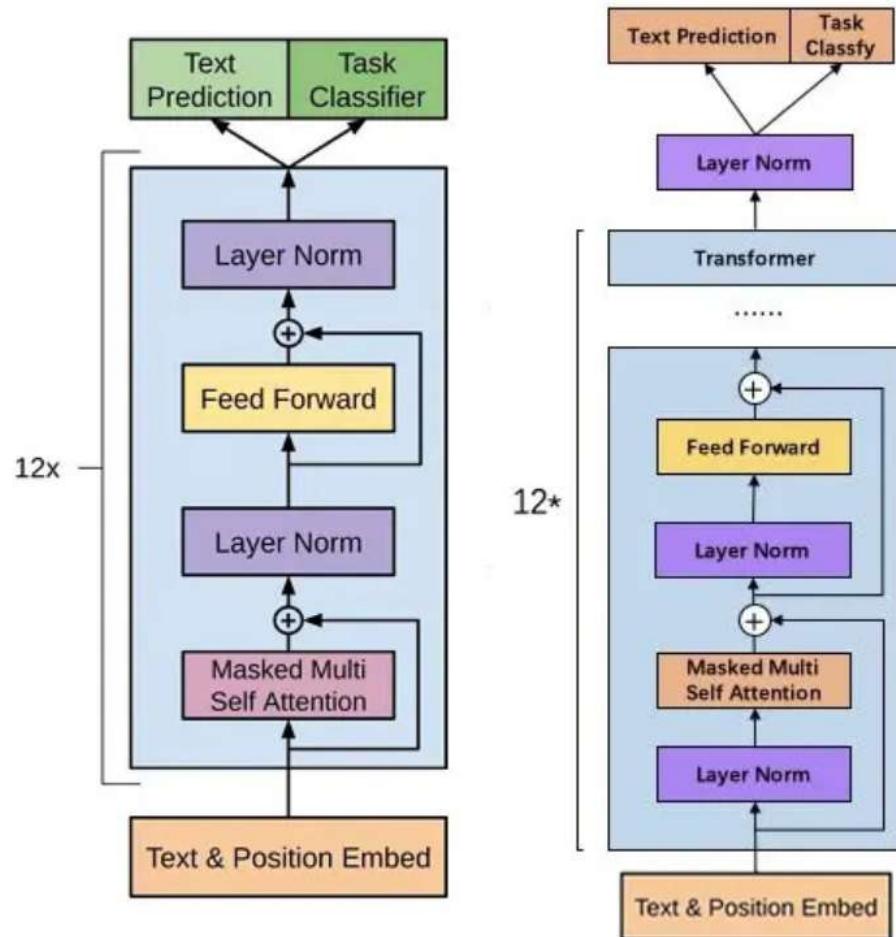


Training method:
autoregressively predict the
next token

Advantages:
generation

Disadvantages:
suboptimal semantic
comprehension

GPT-1/2



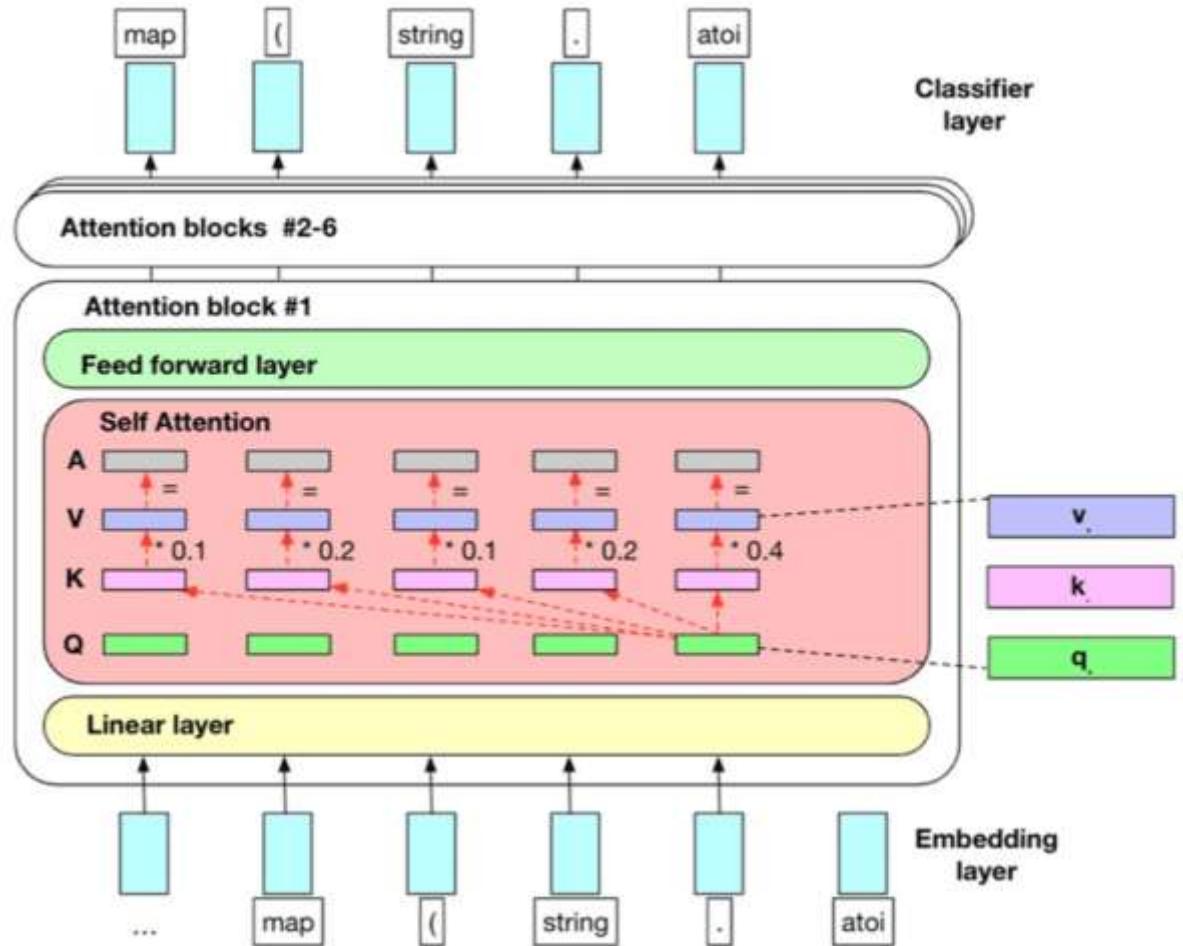
Gpt1:

Training: unsupervised pretrain
+ supervised fine-tuning

Gpt2:

Training: unsupervised pretrain

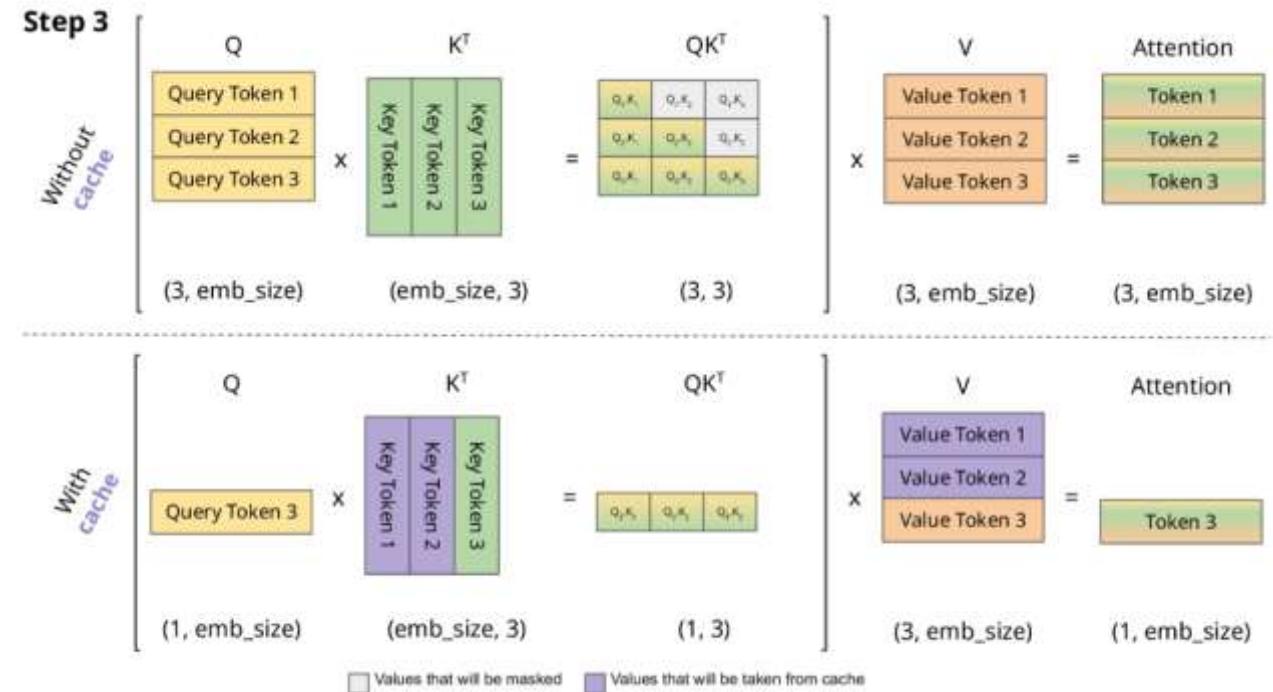
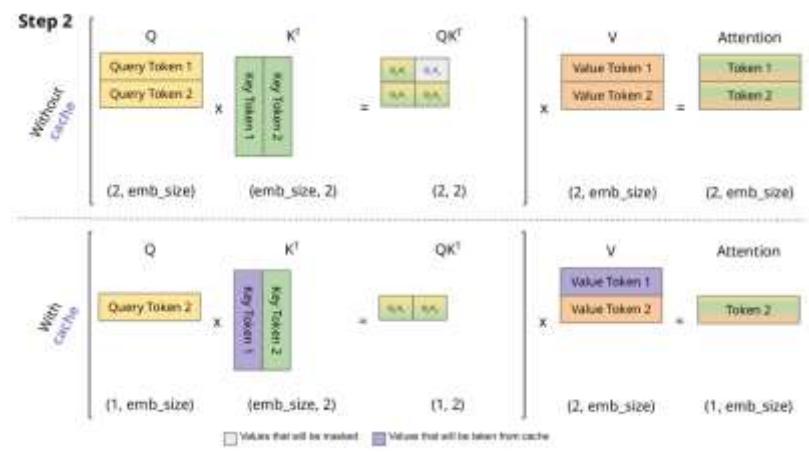
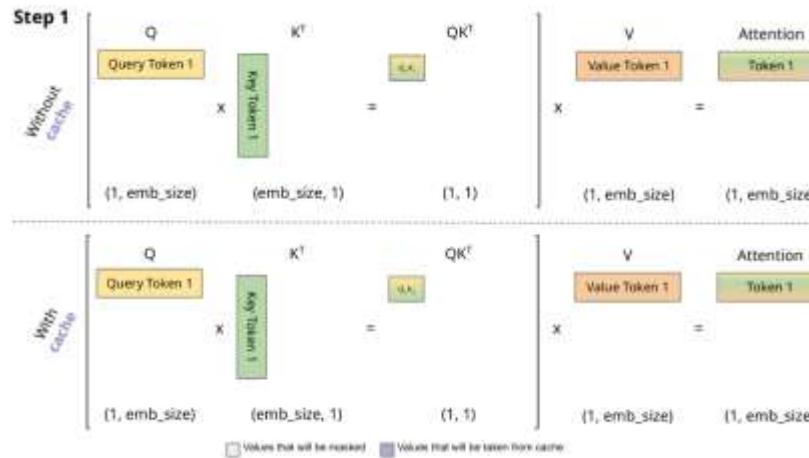
Pretrain



Cross-entropy loss

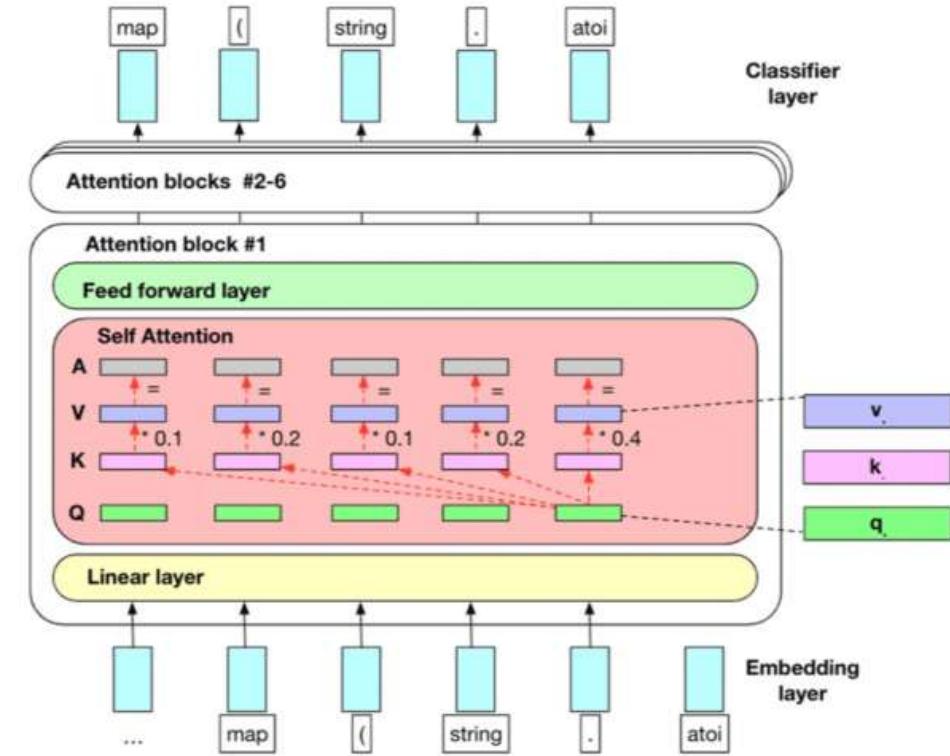
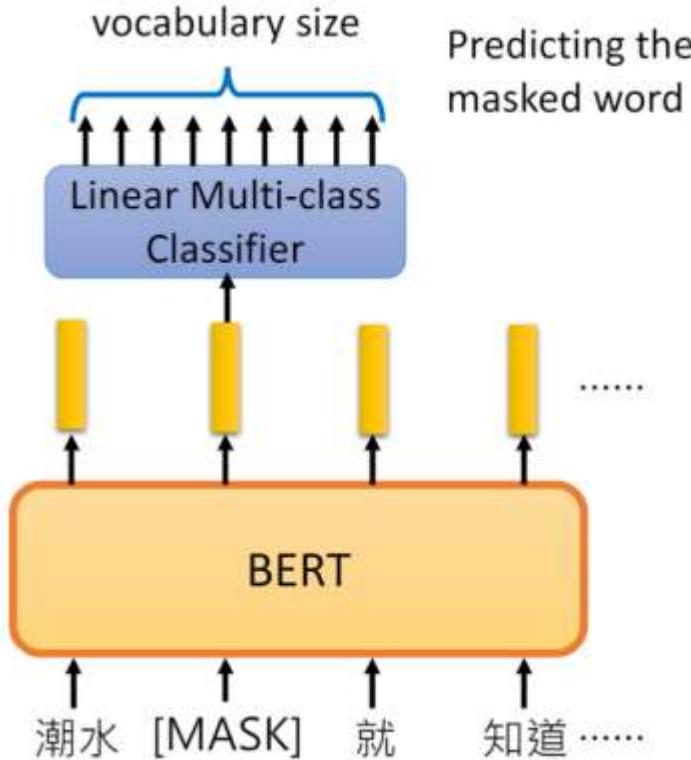
$$L_1(\mathcal{U}) = \sum_i \log P(u_i | u_{i-k}, \dots, u_{i-1}; \Theta)$$

KV cache



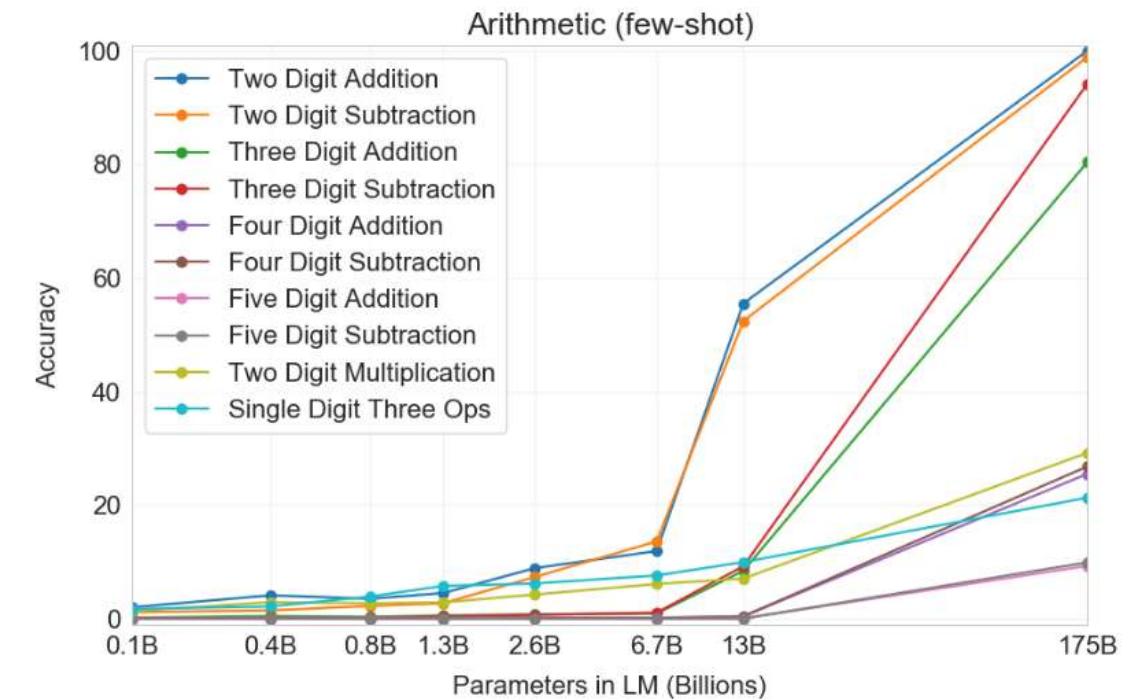
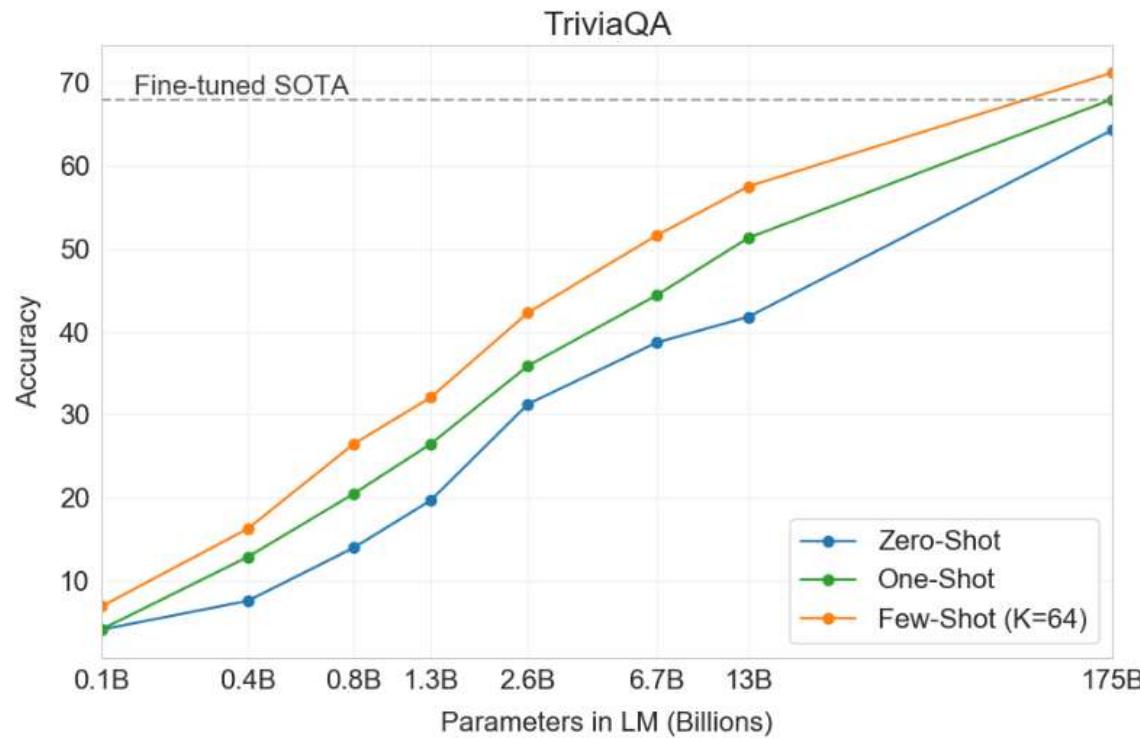
$O(n^2) \rightarrow O(n)$

Scale up——Encoder or Decoder?



- 1、Auto-regression training makes full use of data
- 2、Causal attention enables KV cache

GPT-3——Scaling up



The larger GPT-3's parameter count, the higher its zero-shot/few-shot QA accuracy

arithmetic abilities of GPT-3

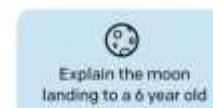
Instruct-GPT(RLHF)

trained to produce outputs that align with human preferences

Step 1

Collect demonstration data, and train a supervised policy.

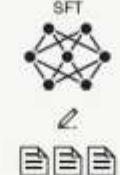
A prompt is sampled from our prompt dataset.



A labeler demonstrates the desired output behavior.



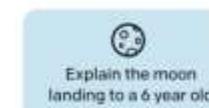
This data is used to fine-tune GPT-3 with supervised learning.



Step 2

Collect comparison data, and train a reward model.

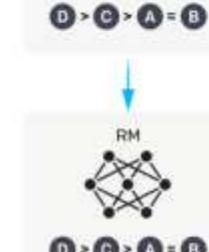
A prompt and several model outputs are sampled.



A labeler ranks the outputs from best to worst.



This data is used to train our reward model.



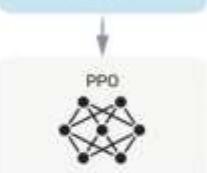
Step 3

Optimize a policy against the reward model using reinforcement learning.

A new prompt is sampled from the dataset.



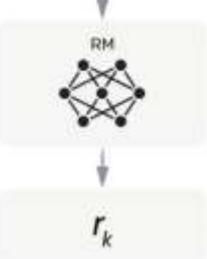
The policy generates an output.



The reward model calculates a reward for the output.



The reward is used to update the policy using PPO.



Reasoning

Cot

The reasoning ability of large language models can be unlocked by simple methods

Standard Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. X

Chain-of-Thought Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had $23 - 20 = 3$. They bought 6 more apples, so they have $3 + 6 = 9$. The answer is 9. ✓

Few-shot cot

(d) Zero-shot-CoT (Ours)

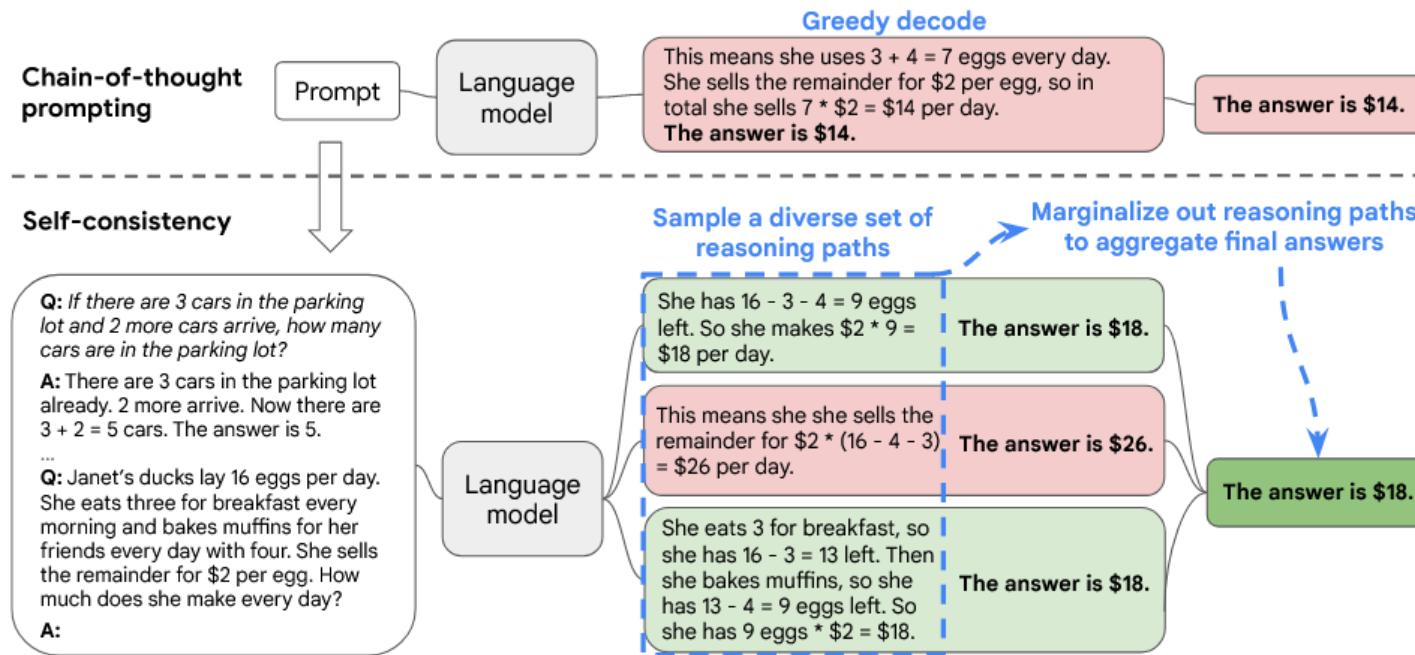
Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A: **Let's think step by step.**

(Output) There are 16 balls in total. Half of the balls are golf balls. That means that there are 8 golf balls. Half of the golf balls are blue. That means that there are 4 blue golf balls. ✓

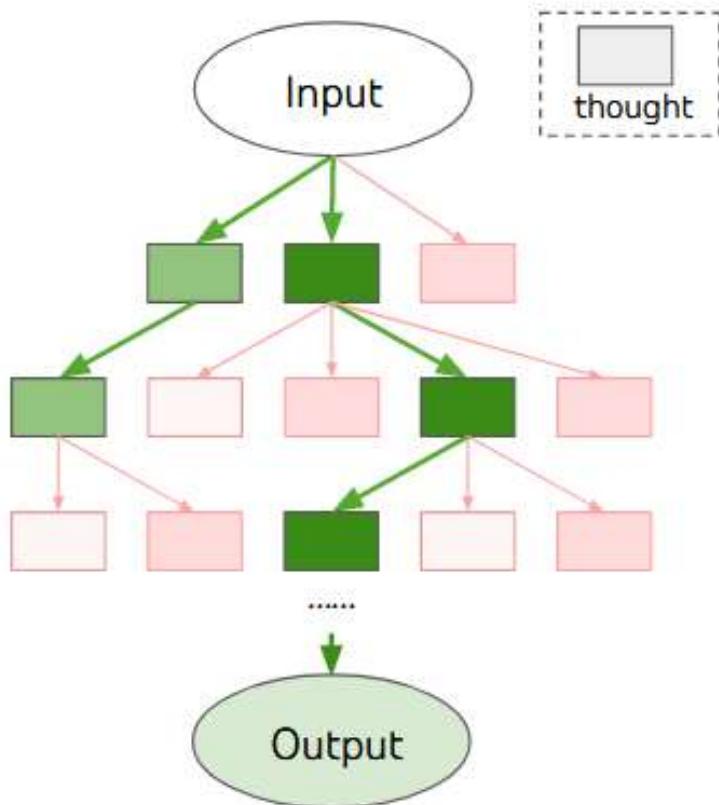
Zero-shot cot

Self-Consistency Cot



Self-consistency leverages the intuition that complex reasoning tasks typically admit multiple reasoning paths that reach a correct answer.

Tot



(d) Tree of Thoughts (ToT)

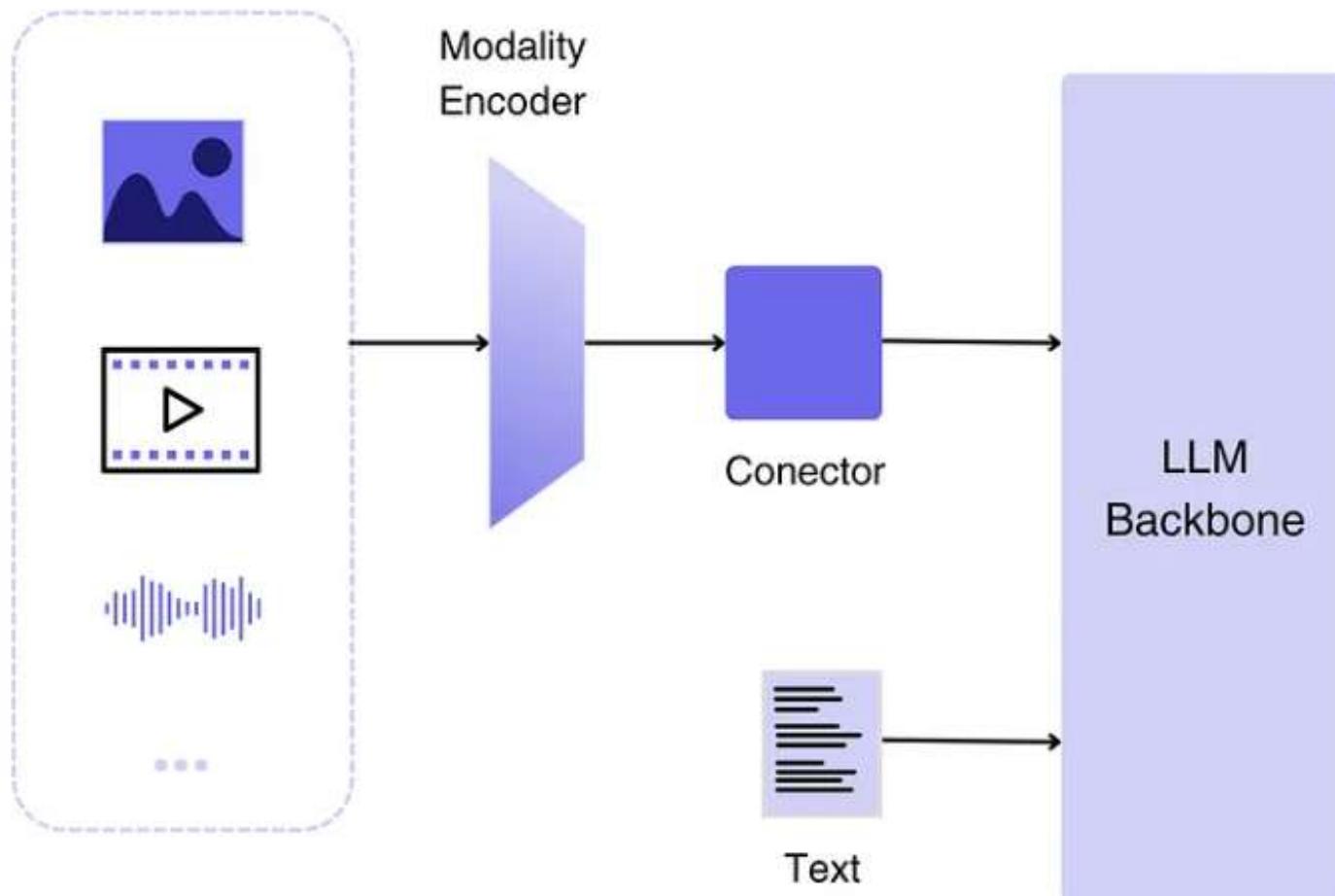
Thought decomposition: Decompose intermediate thinking into coherent based on task characteristics, providing a clear carrier for reasoning.

Thought generator: Generate k diverse candidate thoughts via independent sampling or sequential proposal based on the current reasoning state.

State evaluator: Leverage LLM's autonomous reasoning to assess candidate states.

MLLM

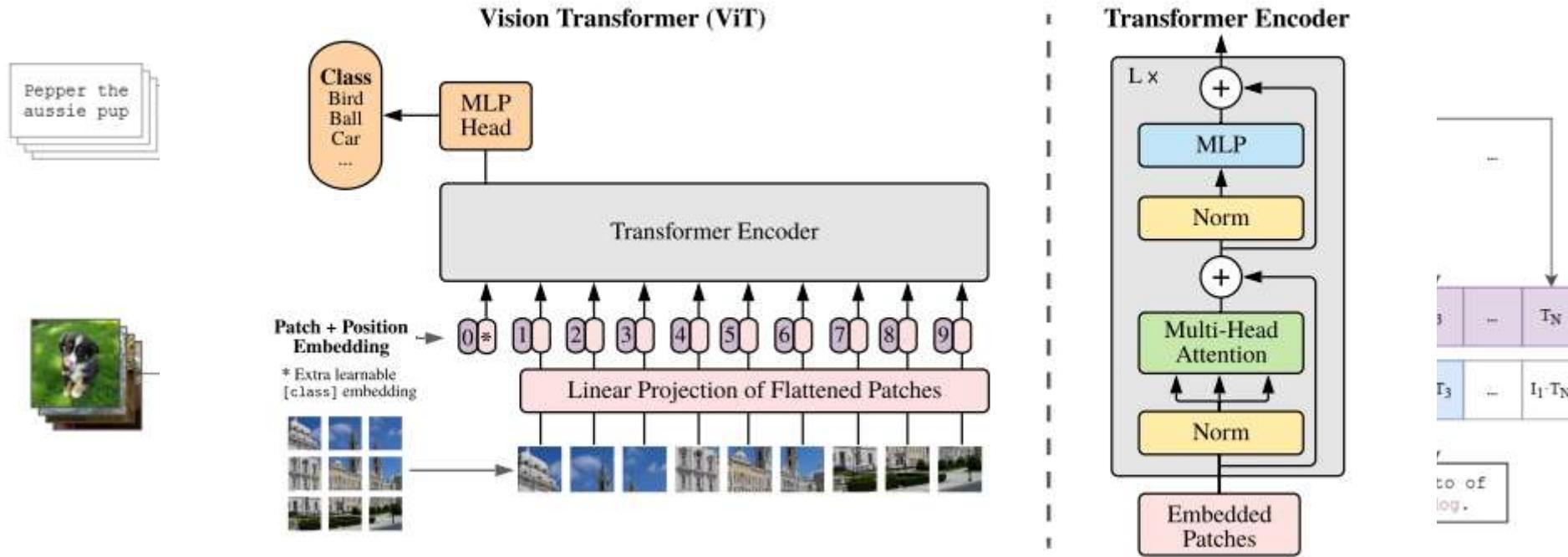
Multimodal Model Architecture



1. Modality encoder
2. Connector
3. LLM

Modality encoder

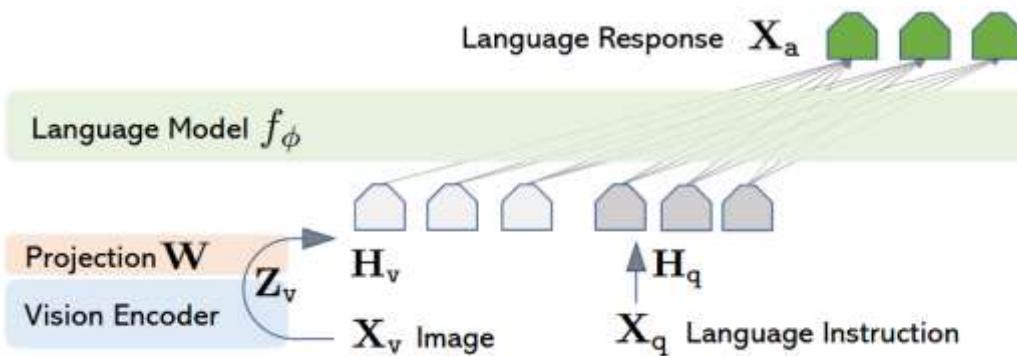
(1) Contrastive pre-training



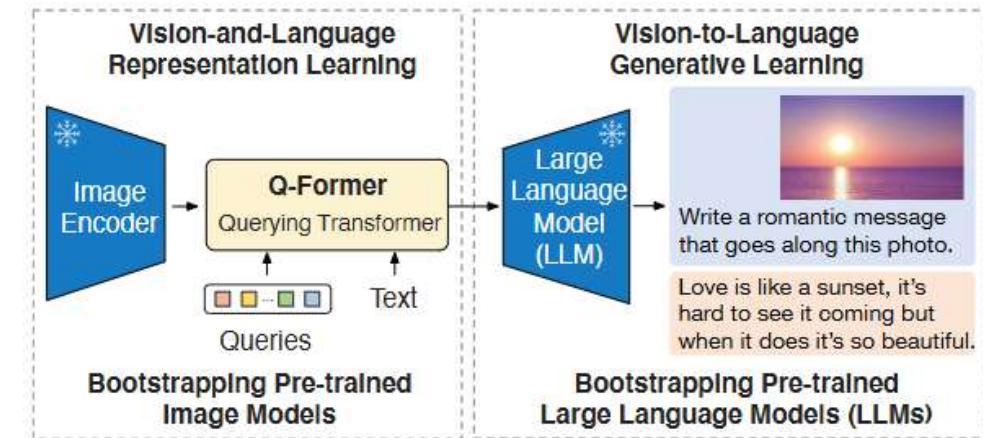
Text-encoder: transformer language model(decoder-only)
Image-encoder: ViT/ResNet

Connector

LLaVA connector: Linear layer

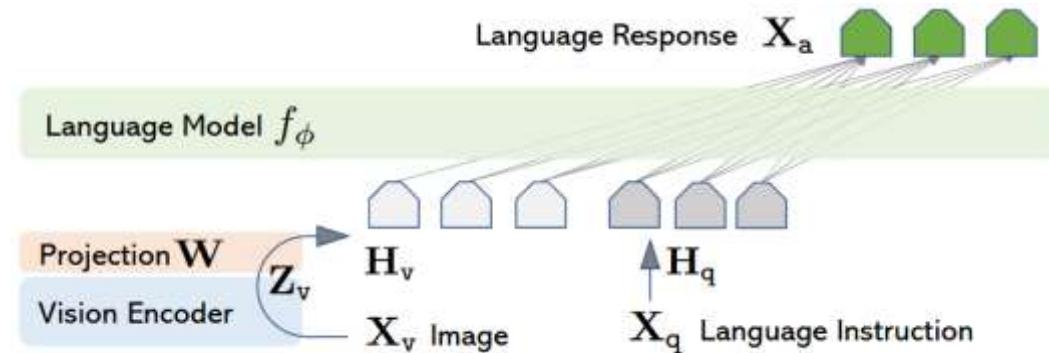
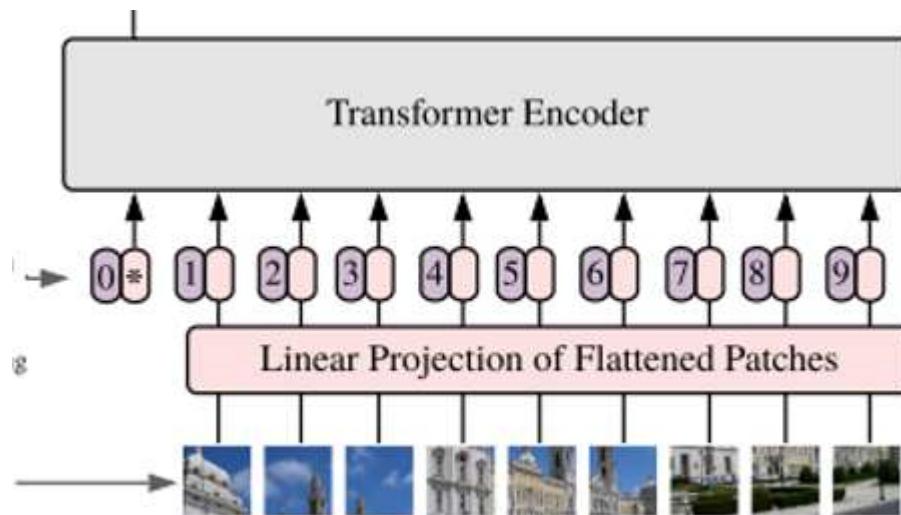


Blip-2 connector: Q-former



Connector

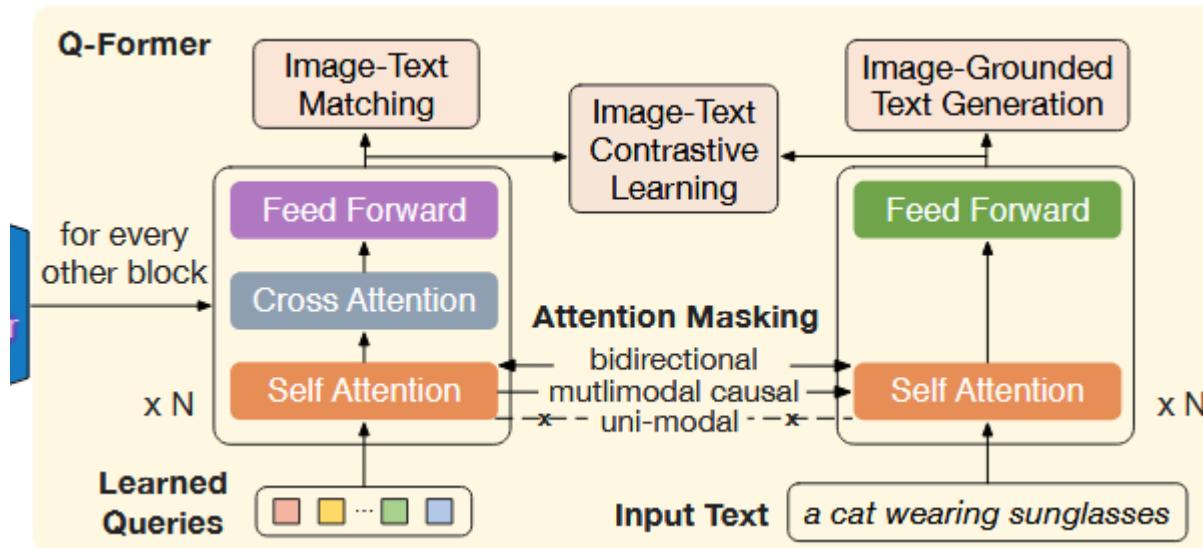
LLaVA connector: Linear layer



Connector (Projector): Aligns ViT Feature Dimension with LLaMA Word Embedding Dimension

Connector

Blip-2 connector: Q-former



- 1、 self-attention: Bert base(pretrained)
- 2、 cross-attention: randomly initialized

ITC: Image-text Contrastive
ITG: Image-text Generation
ITM: Image-Text Matching