基础论文阅读：注意力机制与专家系统

北京科技大学　　汇报人：何开煜

Transformer

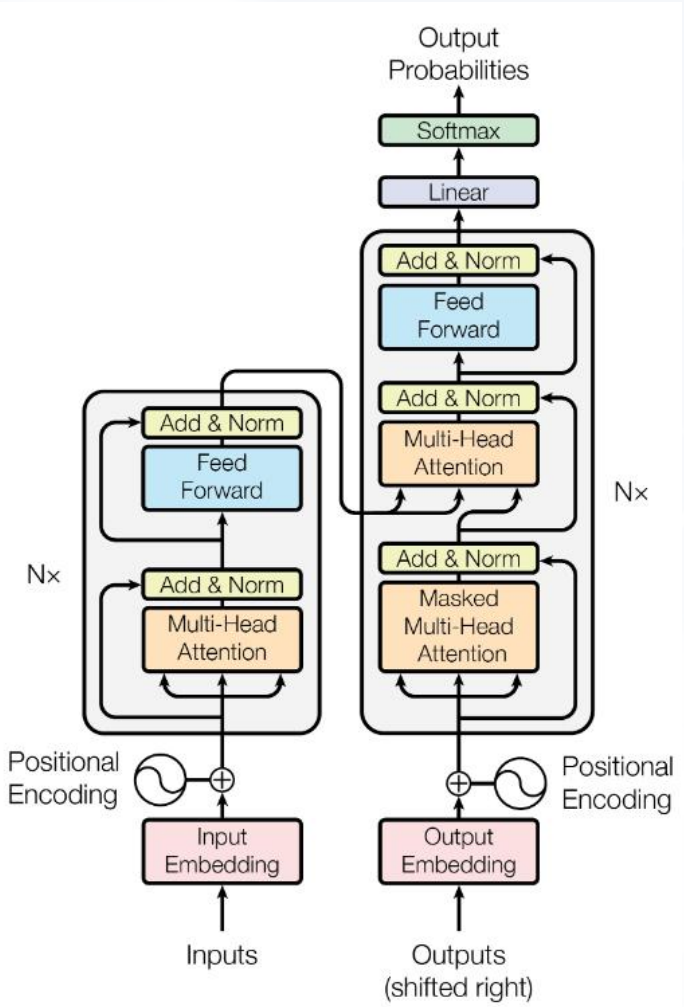


Figure 1: The Transformer - model architecture.

为什么采用自注意力

能够实现并行计算

相对于CNN和RNN能更好处理长距离的依赖关系。

组成部分

Encoder: 可用于处理已知整体序列的问题

衍生模型：BERT

Decoder: 可用于处理序列预测问题，更适合用于生成任务

衍生模型：GPT3、LLaMA、BLOOM

代价

- 相对于RNN,标准自注意力机制的计算复杂度为$O(n^2)$
- 注意力机制难以捕捉序列中的位置信息（存在逆转诅咒）

[1] Attention is All you Need

Transformer

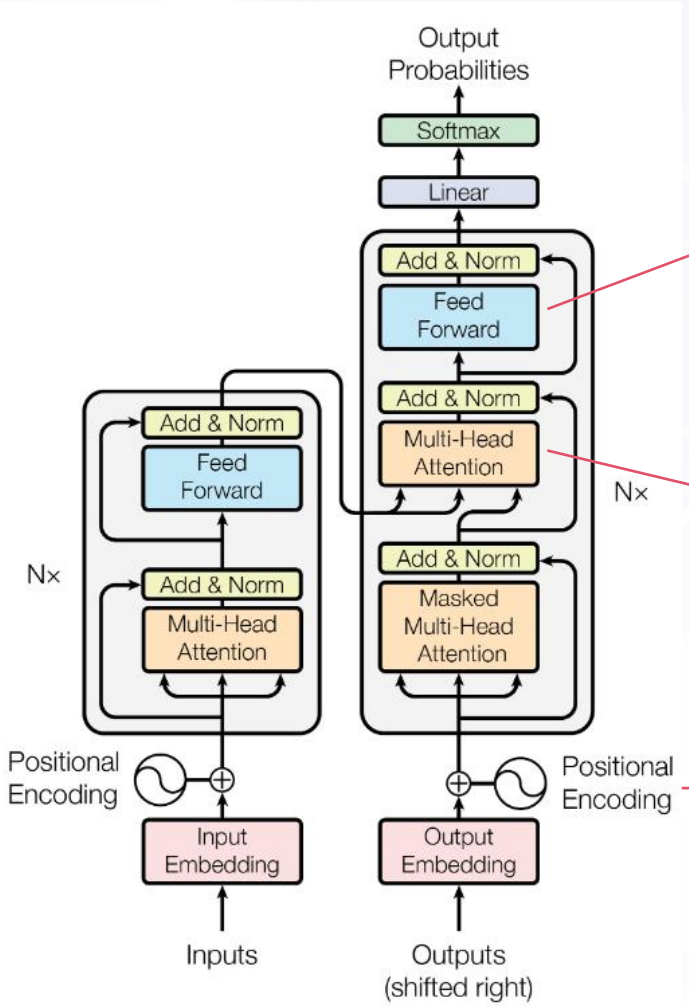


Figure 1: The Transformer - model architecture.

更细致的组成

FFN:

存储模型学习知识的场所

提供非线性变化

实现：两层的感知机

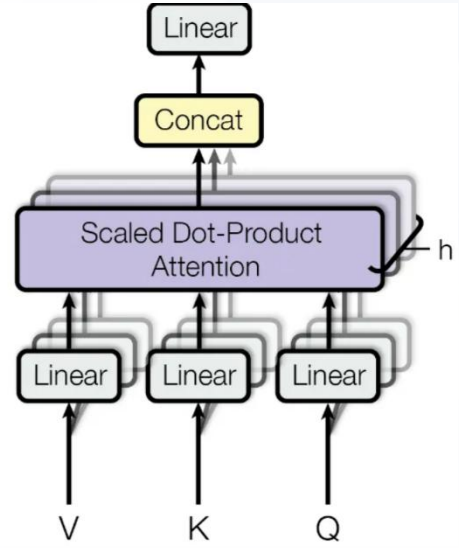
Multi-Head Attention:

获取序列前后的依赖信息，让模型能够关注到与预测位置相关联的信息。

Positional Encoding:

对序列中的数据添加位置的编码信息。

实现：通过添加不同频率的正余弦函数来传递位置信息



Multi-Head Attention

Positional Encoding

Positional Encoding 的类型

- 绝对位置编码

优势：简单且具有较好的可解释性

局限性：可能无法充分表达复杂的位置信息，难以处理长文本

- 相对位置编码

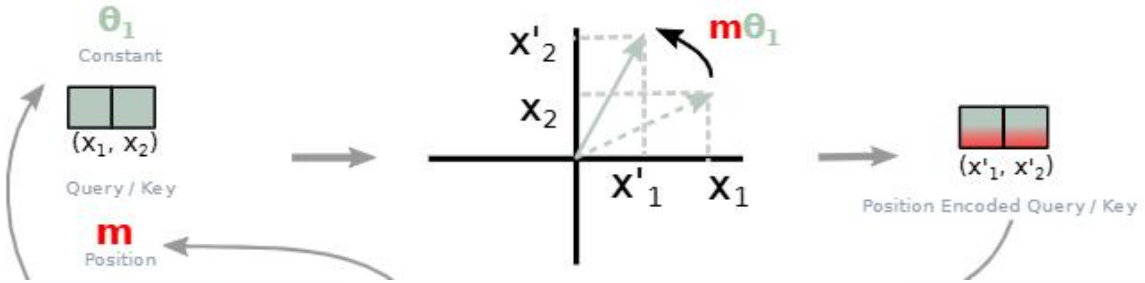
相对绝对位置编码更能捕捉长距离依赖关系

优势：融入了上下文的相对位置信息

- 可学习位置编码

融入了可学习的参数

缺点：需要花费更多的计算资源



相对于绝对位置编码为什么选用相对位置编码？

直觉上认为：序列中相近的部分应该有着更强的相关性。较远距离的token应该有着更少的关联。

RoPE——旋转位置编码

借助旋转矩阵嵌入位置信息

采用绝对位置编码的形式实现了相对位置编码

性能：

- 长期衰减
- 适配线性注意力机制
- 计算高效

应用：

LlaMA 的 k,v

[2] RoFormer: Enhanced Transformer with Rotary Position Embedding

FFN的拓展：专家系统 MoE

MoE架构：

将 FFN 扩充为多个 FFN，作为多个领域的专家，并添加门控系统学习不同序列需要什么样的专家。

Switch Transformers:

指导设计原则：保持浮点操作数不变的前提下，最大化模型的参数量

- 只将token路由到单个专家

专家的容量至少可以减半

简化路由降低通信成本

- 设定专家接收token的容量，token数量大于容量则直接溢出
- 实现可微的平衡负载损失

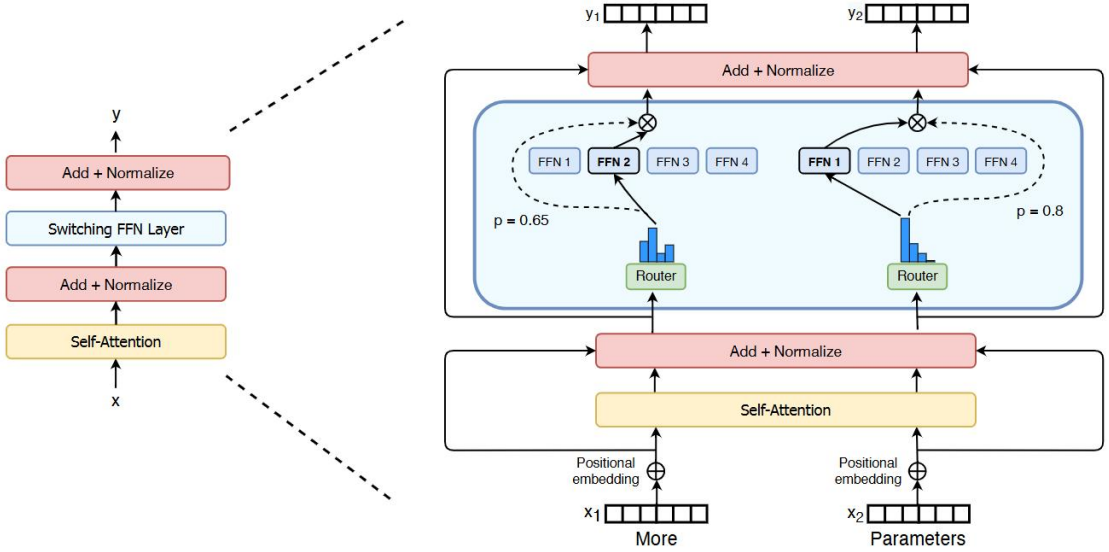
Switch Transformers 为什么会更好

scaling law:

$$L(D,N,P) \approx A * N^{-\alpha} + B * D^{-\beta} + C^{-\gamma}$$

模型的规模越大，模型所得到的损失越低。

经可能的简化模型运算，并扩充模型的参数量。



[3] Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity

FFN的拓展：专家系统MoE

MoE架构的潜在问题：

知识混合性：不同的专家学习的知识难以进行综合。

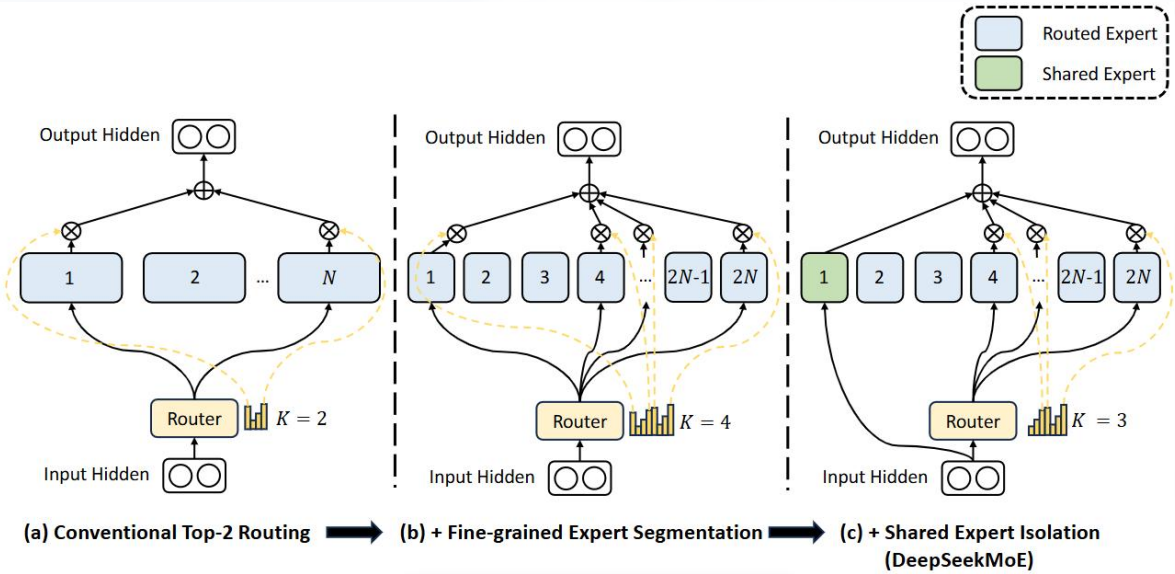
知识冗余：不同的专家学习了相同的知识，影响了专家的专业性

优越性：

其不仅在相同激活参数的前提下有着高于其他模型的性能，还能够从微调中获得收益。

DEEPSEEKMOE:

- 减少专家的维度（容量），扩充其总量和激活的专家数量，使得专家能够保持更高水平的专业性，同时能够有更多样的组合
- 设置共享专家，减少其他专家之间的参数冗余。
- 负载均衡考虑：设计专家级平衡损失和设备级平衡损失



[4] DEEPSEEKMOE: TOWARDS ULTIMATE EXPERT SPECIAL-IZATION IN MIXTURE-OF-EXPERTS LANGUAGE MODELS

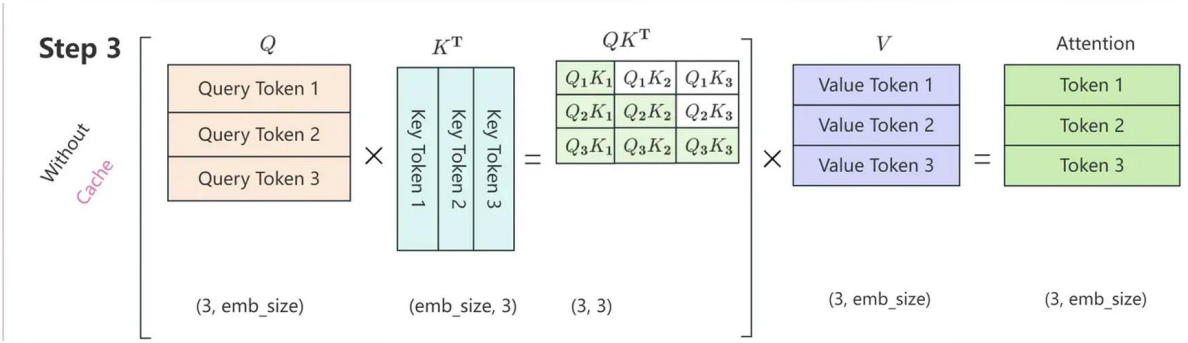
Attentino 的推理优化

KV Cache

KV Cache 是大模型推理性能优化的一个常用技术，通过空间换时间的思想，提高推理性能。

实现：在推理的时候，前面已经生成的 token 不需要与后面的 token 计算 Attention，从而使得前面已经计算的 K 和 V 可以缓存起来。

代价：由于需要将 attention 中的 KV 值存储在显卡中，因此KV Cache 可能是显存刺客。



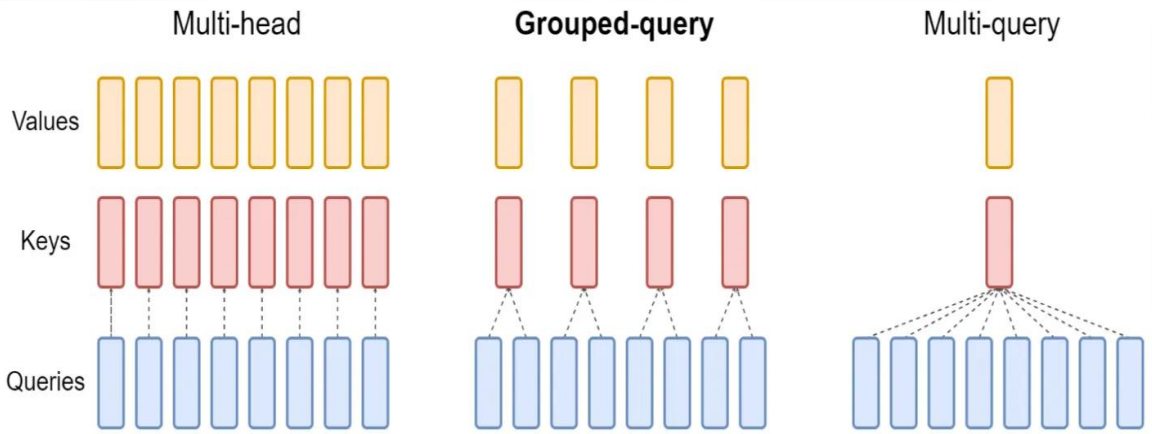
较少KV Cache的显存占用实现更长上下文和更大batch size的推理

MQA

直接让所有的 Attention Head 共享同一个K、V

GQA

对Q进行分组通过一对K、V



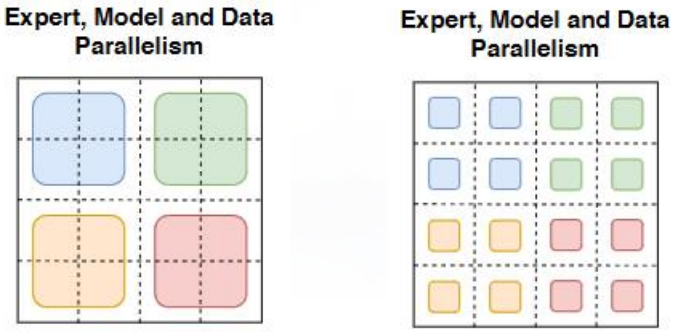
[5] GQA: Training Generalized Multi-Query Transformer Models from Multi-Head Checkpoints

预训练

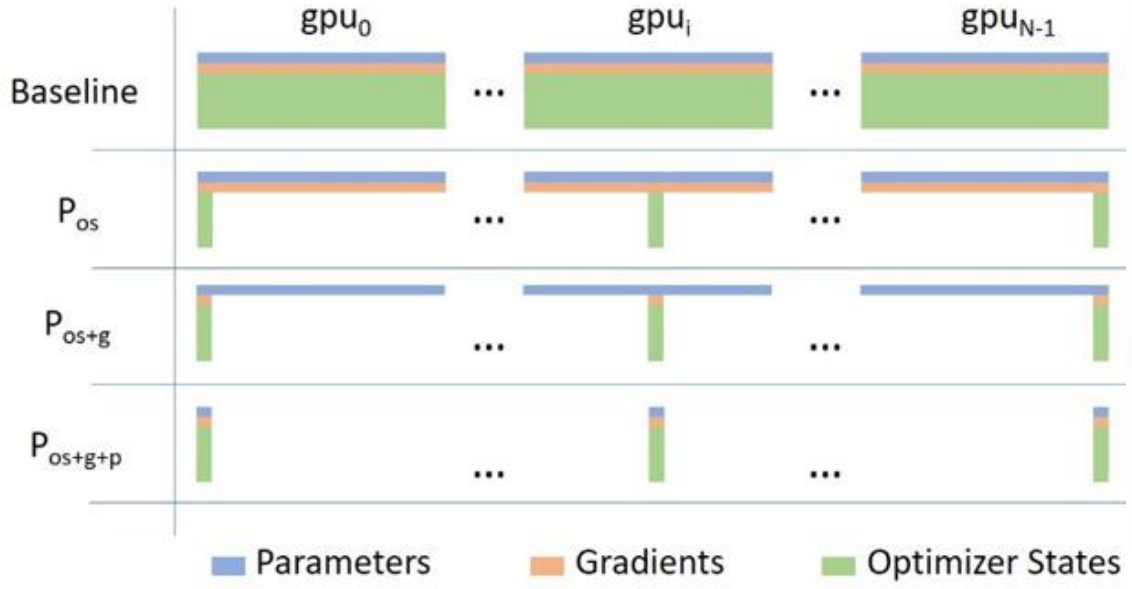
3D 训练模式: DP+PP+TP

- 数据并行 DP：将不同的训练数据放置在不同的GPU
- 流水线并行 PP：将模型不同的层放置在不同的GPU
- 张量并行 TP：将张量切片放置在不同的GPU

- 专家并行：将不同的专家放置在不同的显卡上



Zero的三种模式

在实现数据并行的基础上（Zero 0），对模型优化器（Zero 1）、梯度（Zero 2）、模型参数（Zero 3）进行切片进行训练。

通过通信时间开销换取更小的显存占用