

轻量化微调

《大语言模型》编写团队：唐天一

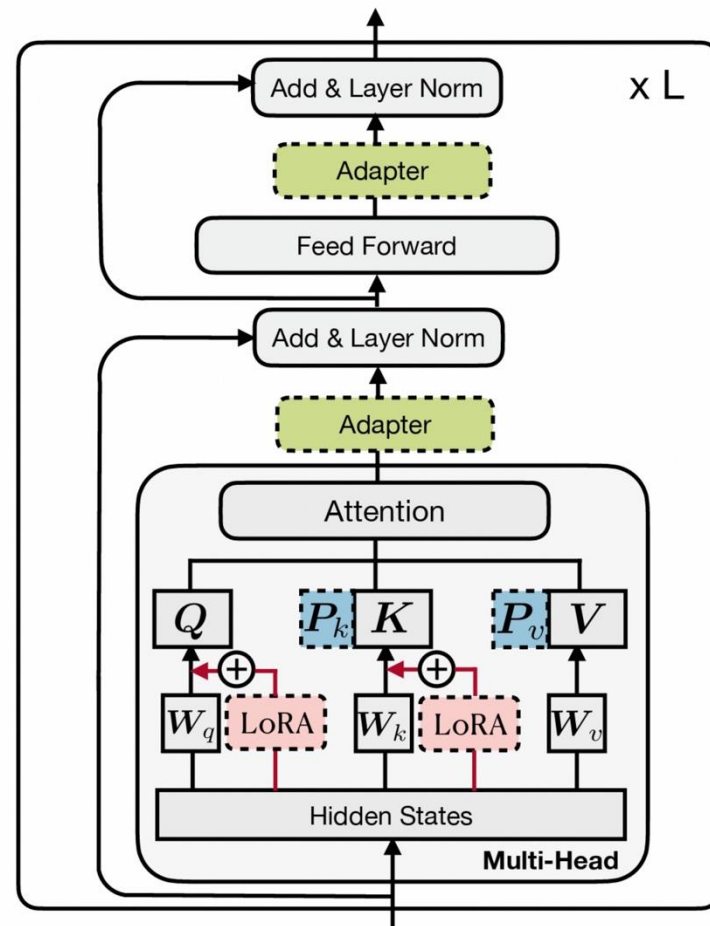
轻量化微调

➤ 训练时每张GPU显存占用计算公式：

$$\underbrace{\frac{16\phi}{N}}_{\text{模型与优化器}} + \underbrace{2LBTH + 12BTV}_{\text{激活值}} + \underbrace{6}_{\text{其他}}$$

➤ 轻量化微调的目的

- 减少模型训练参数量，从而降低显存占用
- 同时尽可能接近全量微调的性能

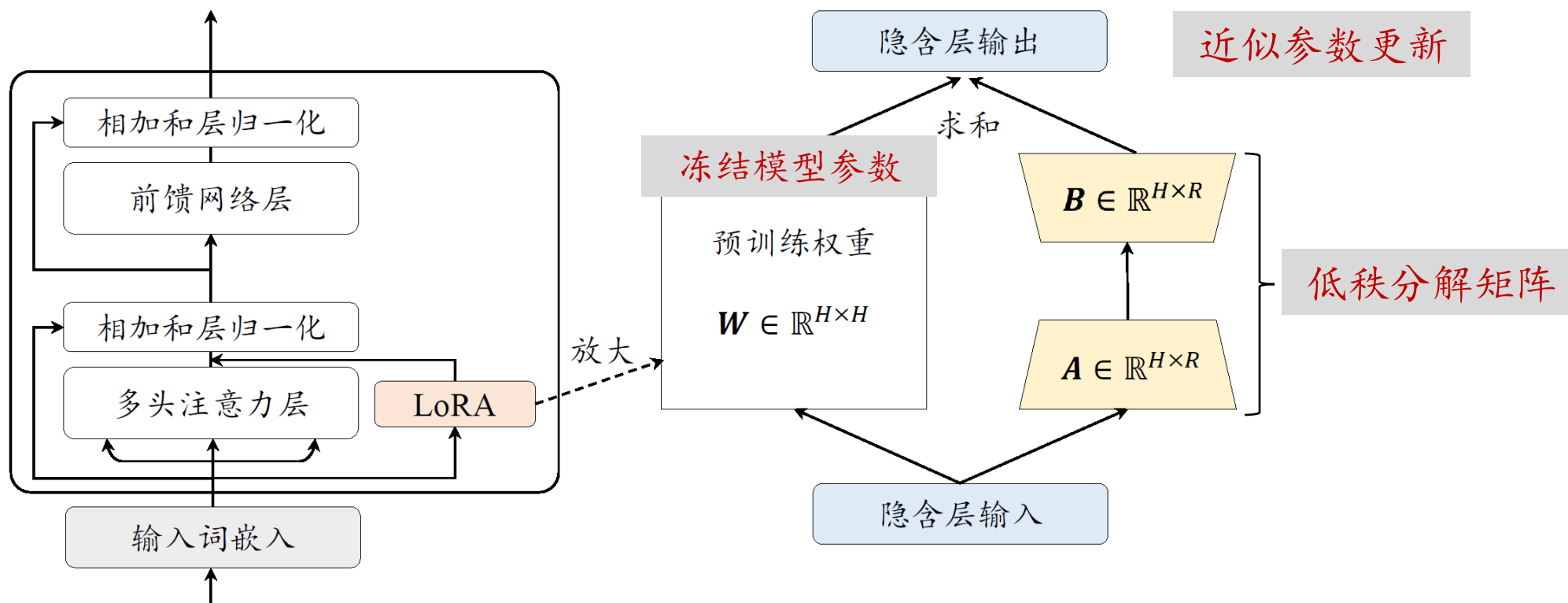


冻结LLM参数，只微调极少量额外参数

低秩适配微调方法（LoRA）

➤ LoRA 更新参数 W 过程如下

➤ $W \leftarrow W + \Delta W = W + A \cdot B^T$

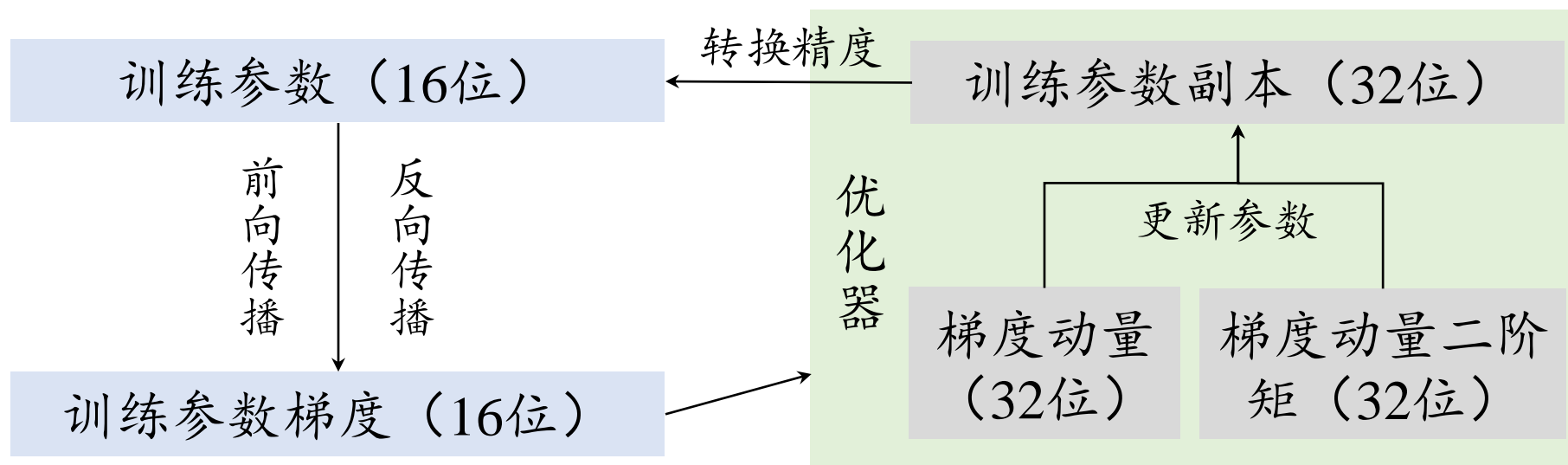


低秩适配微调方法（LoRA）

➤ LoRA 微调显存占用情况

➤ 模型： $2P + 2P_{\text{LoRA}}$ ，梯度： $2P_{\text{LoRA}}$ ，优化器： $12P_{\text{LoRA}}$

➤ 从 $16P$ 减为 $2P + 16P_{\text{LoRA}}$



低秩适配微调方法（LoRA）

- LoRA 用于注意力层的线性变换 W^K 和 W^V
 - $\phi_{\text{LoRA}} = 2 * 2LHR + 2 * 2LHR = 8LHR$ (R通常取16)
- 以 LLaMA 7B 为例, $\phi \approx 6.7 \times 10^9$, $\phi_{\text{LoRA}} \approx 1.7 \times 10^7$, $\phi_{\text{LoRA}} \ll \phi$
 - 模型和优化器占用从 $16P$ 降至 $2P$
 - 3090 24G 可以微调 7B 模型
- QLoRA: 量化参数矩阵, 用 4 比特存储模型参数
 - 模型占用从 $2P$ 降至 $0.5P$
 - A6000 48G 可以微调 65B 模型

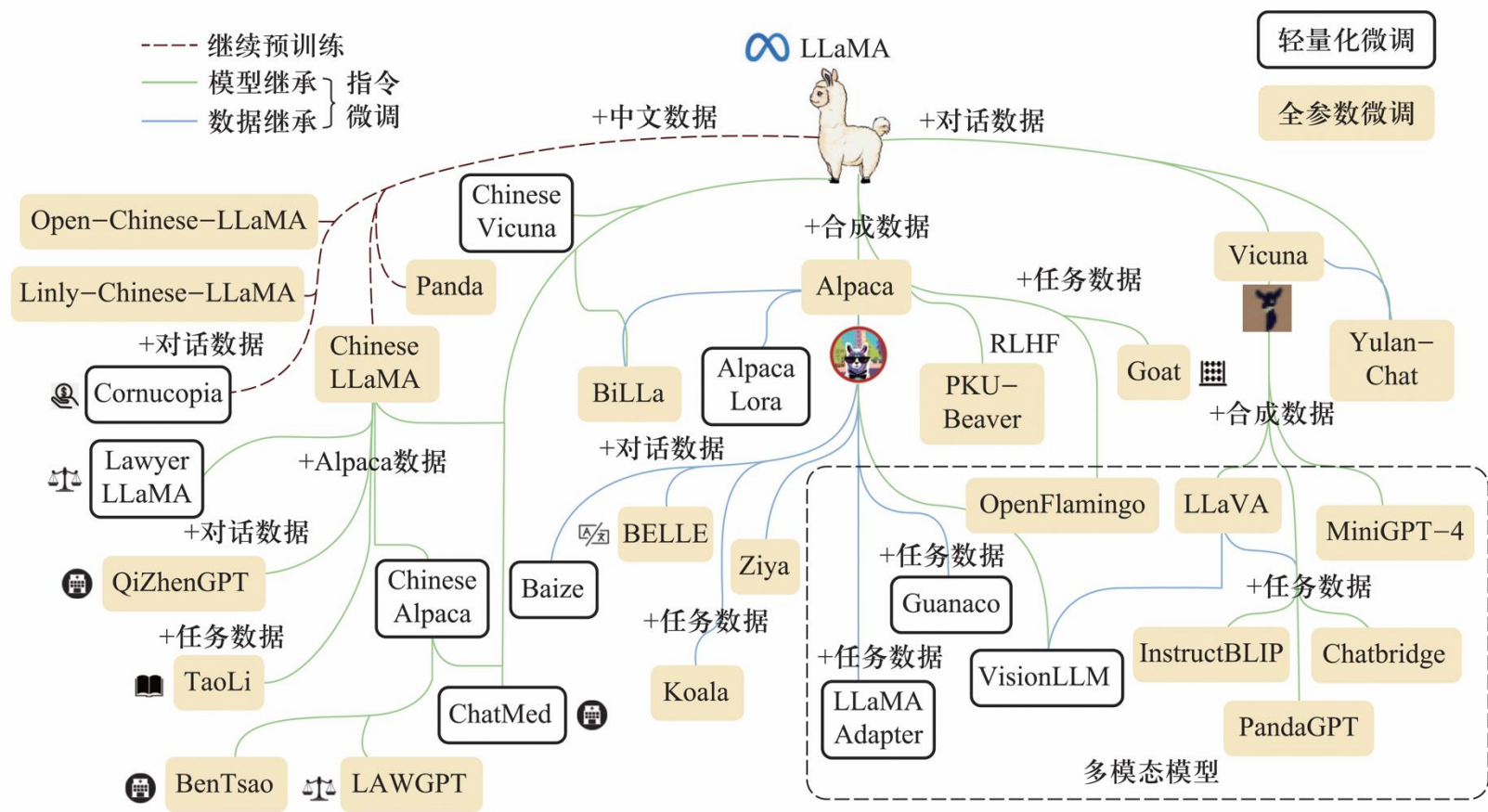
- LoRA微调 Alpaca-52K 所需的 A800 (80 G) 数量、批次大小和微调时间
 - 使用数据并行、ZeRO-3、BF16 和激活重计算技术

模型	GPU 数量	批次大小	时间
LLaMA (7 B)	1	16	2.3 h
LLaMA (13 B)	1	8	3.8 h
LLaMA (33 B)	1	1	10.2 h
LLaMA (65 B)	2	1	26.0 h

低秩适配微调方法（LoRA）



➤ SFT 以及 LoRA 在下游应用中的适配



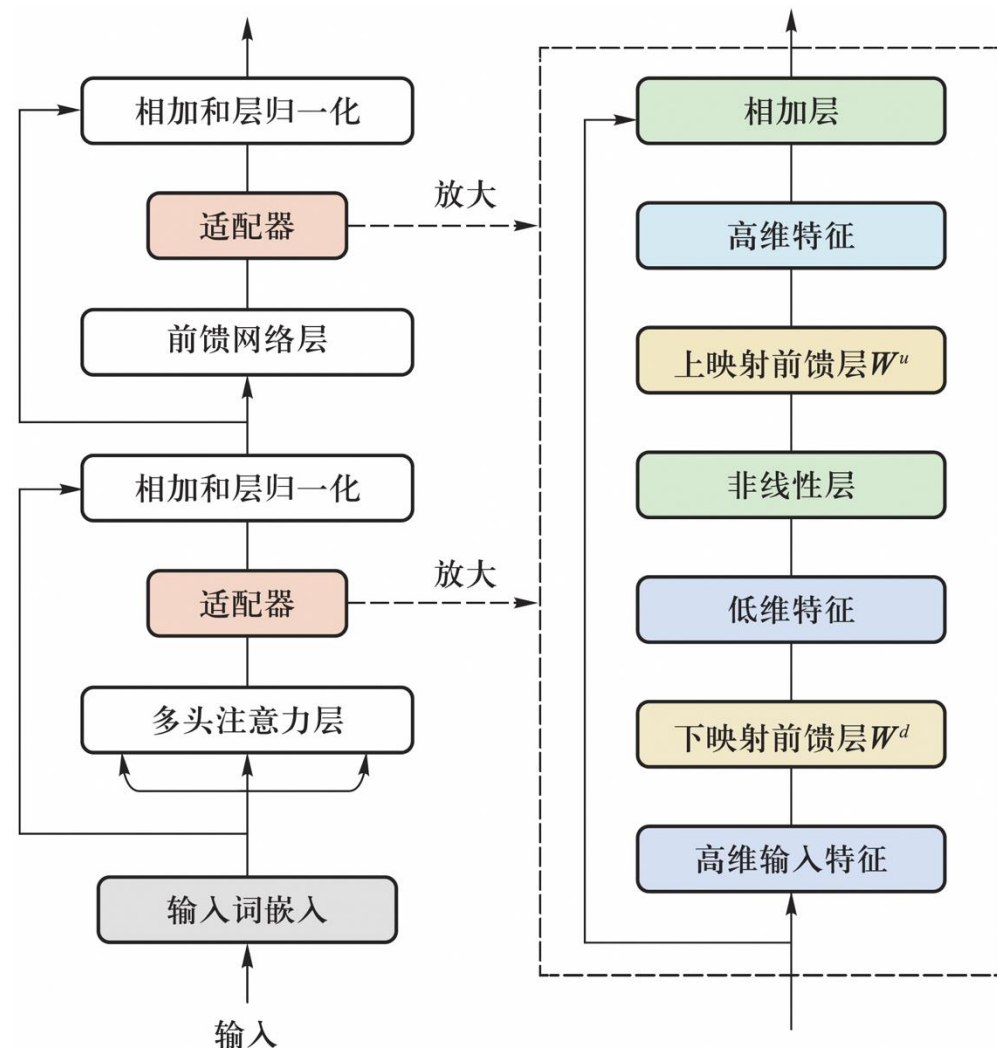
其他轻量化微调方法

➤ 适配器微调 (Adaptor tuning)

➤ 引入小型神经网络模块

➤ $\mathbf{h} = \mathbf{h} + \sigma(\mathbf{h} \cdot \mathbf{W}^d) \cdot \mathbf{W}^u$

➤ $\mathbf{W}^d \in \mathbb{R}^{H \times R}$, $\mathbf{W}^u \in \mathbb{R}^{R \times H}$, $R \ll H$

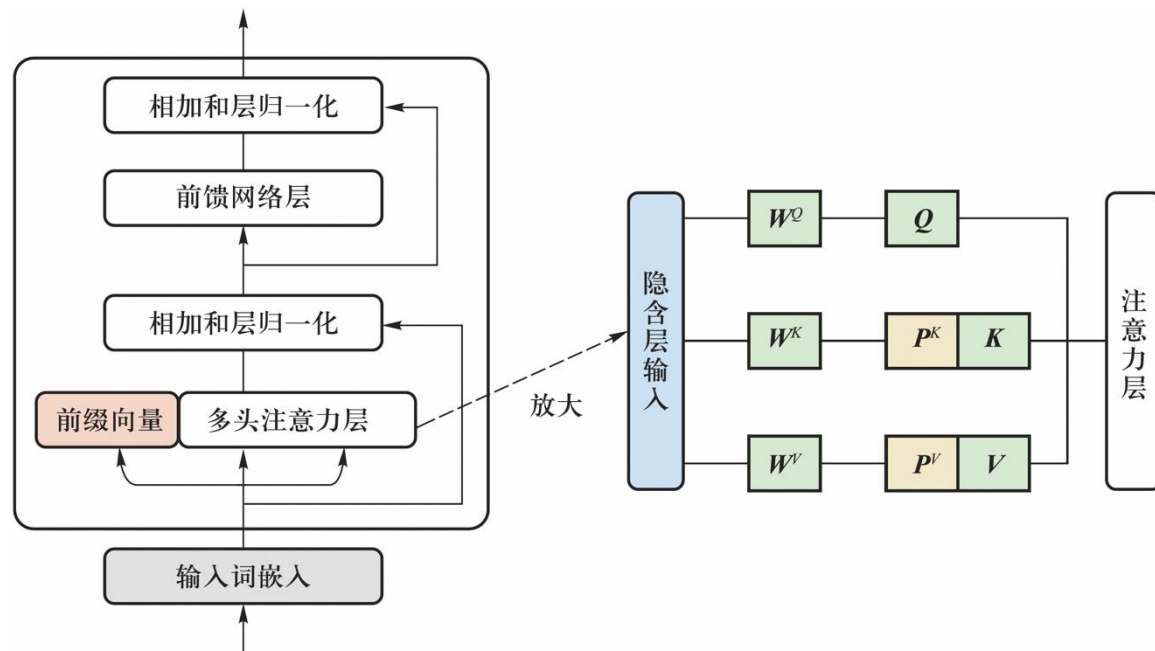


其他轻量化微调方法

➤ 前缀微调 (Prefix tuning)

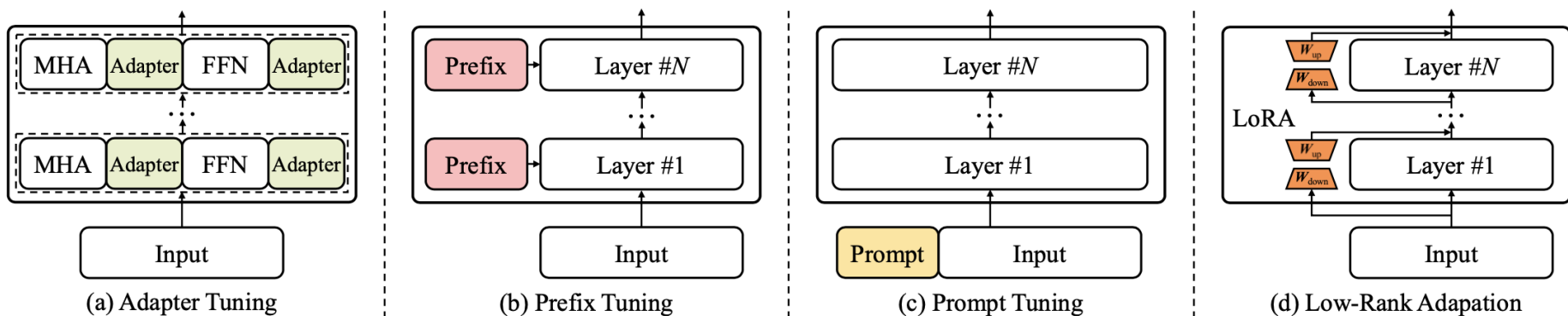
➤ 在多头注意力层中添加前缀参数

➤ $\text{Attention}(XW^Q, P^K \oplus XW^K, P^V \oplus XW^V)P^K, P^V \in \mathbb{R}^{L \times H}$, L 常取10-100



其他轻量化微调方法

➤ 不同轻量化方法的对比



➤ 适配器微调：每层都加一个小网络，加在两个核心组件后面

➤ 前缀微调：每层输入都加入prompt向量

➤ 提示微调：只在输入层加入prompt向量

➤ LoRA 微调：低秩分解学习权重增量部分



谢谢