1 目录

2 大模型基础

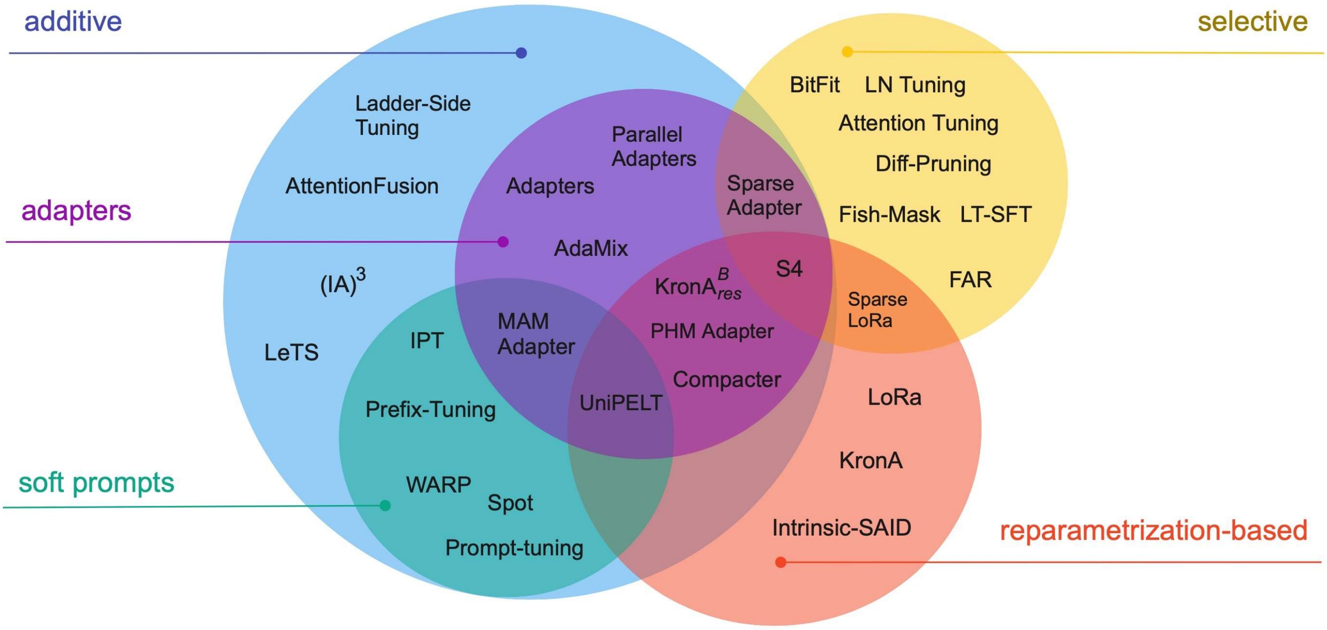
2.1 构建框架

2.2 预训练

(Generative Pre-Training), 由多层Transformer 组成的单向语言模型，主要分为输入层，编码层和输出层三部分。

2.3 有监督微调

有监督微调（Supervised Finetuning, SFT）又称指令微调（Instruction Tuning），是指在已经训练好的语言模型的基础上，通过使用有标注的特定任务数据进行进一步的微调，从而使得模型具备遵循指令的能力。经过海量数据预训练后的语言模型虽然具备了大量的“知识”，但是由于其训练时的目标仅是进行下一个词的预测，此时的模型还不能够理解并遵循人类自然语言形式的指令。



微调技术综述：

```
Scaling Down to Scale Up: A Guide to Parameter-Efficient Fine-Tuning
https://arxiv.org/pdf/2303.15647.pdf
```

2.3.1 BitFit

```
只调节神经网络的bias参数
```

$$\mathrm{Q}^{m,\ell}(\mathrm{x}) = \mathrm{W}_q^{m,\ell}\mathrm{x} + \mathrm{b}_q^{m,\ell}$$
$$\mathrm{K}^{m,\ell}(\mathrm{x}) = \mathrm{W}_k^{m,\ell}\mathrm{x} + \mathrm{b}_k^{m,\ell}$$
$$\mathrm{V}^{m,\ell}(\mathrm{x}) = \mathrm{W}_v^{m,\ell}\mathrm{x} + \mathrm{b}_v^{m,\ell}$$

Where x is the output of the former encoder layer (for the first encoder layer x is the output of the embedding layer). These are then combined using an attention mechanism that does not involve new parameters:

$$\mathrm{h}_1^\ell = att\big(\mathrm{Q}^{1,\ell},\mathrm{K}^{1,\ell},\mathrm{V}^{1,\ell},\ldots,\mathrm{Q}^{m,\ell},\mathrm{K}^{m,\ell},\mathrm{V}^{m,l}\big)$$

and then fed to an MLP with layer-norm (LN):

$$\mathrm{h}_2^\ell = \mathrm{Dropout}\big(\mathrm{W}_{m_1}^\ell \cdot \mathrm{h}_1^\ell \;+\; \mathrm{b}_{m_1}^\ell\big) \quad (1)$$
$$\mathrm{h}_3^\ell = \mathrm{g}_{LN_1}^\ell \odot \frac{(\mathrm{h}_2^\ell + \mathrm{x}) - \mu}{\sigma} + \mathrm{b}_{LN_1}^\ell \quad (2)$$
$$\mathrm{h}_4^\ell = \mathrm{GELU}\big(\mathrm{W}_{m_2}^\ell \cdot \mathrm{h}_3^\ell \;+\; \mathrm{b}_{m_2}^\ell\big) \quad (3)$$
$$\mathrm{h}_5^\ell = \mathrm{Dropout}\big(\mathrm{W}_{m_3}^\ell \cdot \mathrm{h}_4^\ell \;+\; \mathrm{b}_{m_3}^\ell\big) \quad (4)$$
$$\mathrm{out}^\ell = \mathrm{g}_{LN_2}^\ell \odot \frac{(\mathrm{h}_5^\ell + \mathrm{h}_3^\ell) - \mu}{\sigma} + \mathrm{b}_{LN_2}^\ell \quad (5)$$

论文：

```
BitFit: Simple Parameter-efficient Fine-tuning for Transformer-based Masked Langu
age-models
https://arxiv.org/pdf/2106.10199v2.pdf
```

代码：

```
num_param = 0
for name, param in model.named_parameters():
    if "bias" not in name:
        param.requires_grad = False
    else:
        num_param += param.numel()
num_param
```

2.3.2 Prompt-Tuning

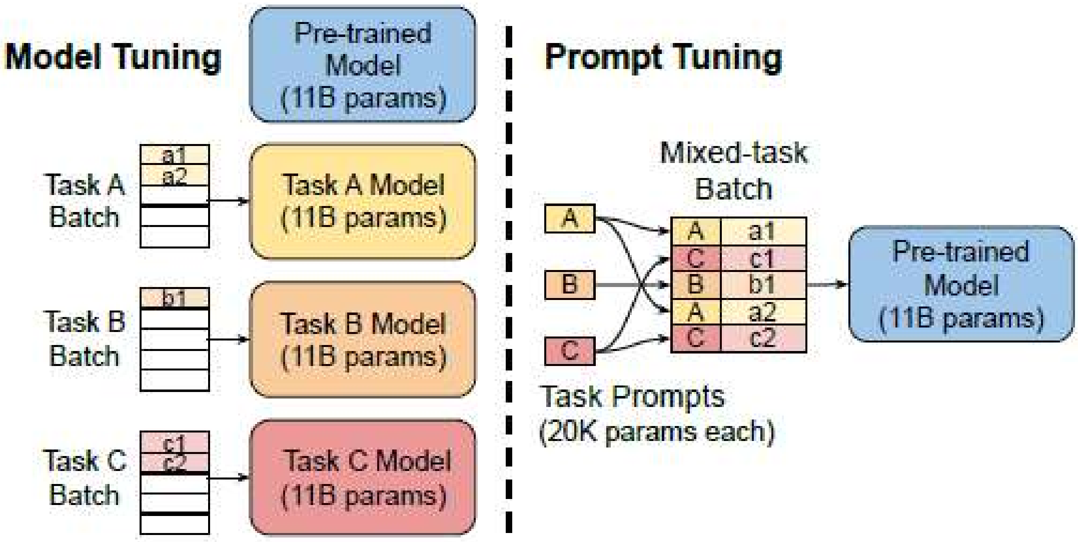


Figure 2: **Model tuning** requires making a task-specific copy of the entire pre-trained model for each downstream task and inference must be performed in separate batches. **Prompt tuning** only requires storing a small task-specific prompt for each task, and enables mixed-task inference using the original pre-trained model. With a T5 "XXL" model, each copy of the tuned model requires 11 billion parameters. By contrast, our tuned prompts would only require 20,480 parameters per task—a reduction of *over five orders of magnitude*—assuming a prompt length of 5 tokens.

论文:

```
The Power of Scale for Parameter-Efficient Prompt Tuning
https://arxiv.org/pdf/2104.08691.pdf
```

算法原理:

$$\hat{Y} = argmax_Y Pr_{\theta,\theta_p}(Y|[P;X])$$

- θ model parameters, θ_p prompt 参数
- Y output, a sequence of tokens
- X input, a sequence of tokens
- P prompt, a series of tokens prepended to the input

2.3.3 P-Tuning

论文:

```
GPT Understands, Too
https://arxiv.org/pdf/2103.10385.pdf
P-Tuning v2: Prompt Tuning Can Be Comparable to Fine-tuning Universally Across Sc
ales and Tasks
https://arxiv.org/pdf/2110.07602.pdf
```

2.3.4 Prefix-Tuning

论文:

```
Prefix-Tuning: Optimizing Continuous Prompts for Generation
https://arxiv.org/pdf/2101.00190.pdf
```

2.3.5 Lora

论文:

```
LORA: LOW-RANK ADAPTATION OF LARGE LANGUAGE MODELS
https://arxiv.org/pdf/2106.09685.pdf
```

2.3.6 IA3

```
Few-Shot Parameter-Efficient Fine-Tuning is Better and Cheaper than In-Context Le
arning
https://arxiv.org/pdf/2205.05638.pdf
```

2.3.7 Adapter

论文:

```
Parameter-Efficient Transfer Learning for NLP
https://arxiv.org/pdf/1902.00751.pdf
```

2.4 强化学习

2.4.1 奖励模型

```
https://zhuanlan.zhihu.com/p/595579042
```

2.4.2 RLHF

3 扩散模型

4 NLP 任务

5 视觉

6 模型训练