微调（Fine-tuining）

- 一个神经网络一般可以分两部分：特征抽取层 + 任务层（分类等）
- 微调指预先在原始数据集学习好特征提取模型，复制应用于目标数据集上（特征提取部分网络结构相同，参数初始化应用前者训练好的参数），之后用目标数据集训练新的任务层，并调整特征提取层参数

1.**训练**

- 是一个目标数据集的正常训练任务，但使用更强的正则化：
  - 更小的学习率
  - 更少的数据迭代（这里的正则化，我的理解是相较从 0 训练更难 over fitting）
  - 原数据集远复杂于目标数据集，通常微调效果才更好（百倍以上）

2.**重用分类器权重**

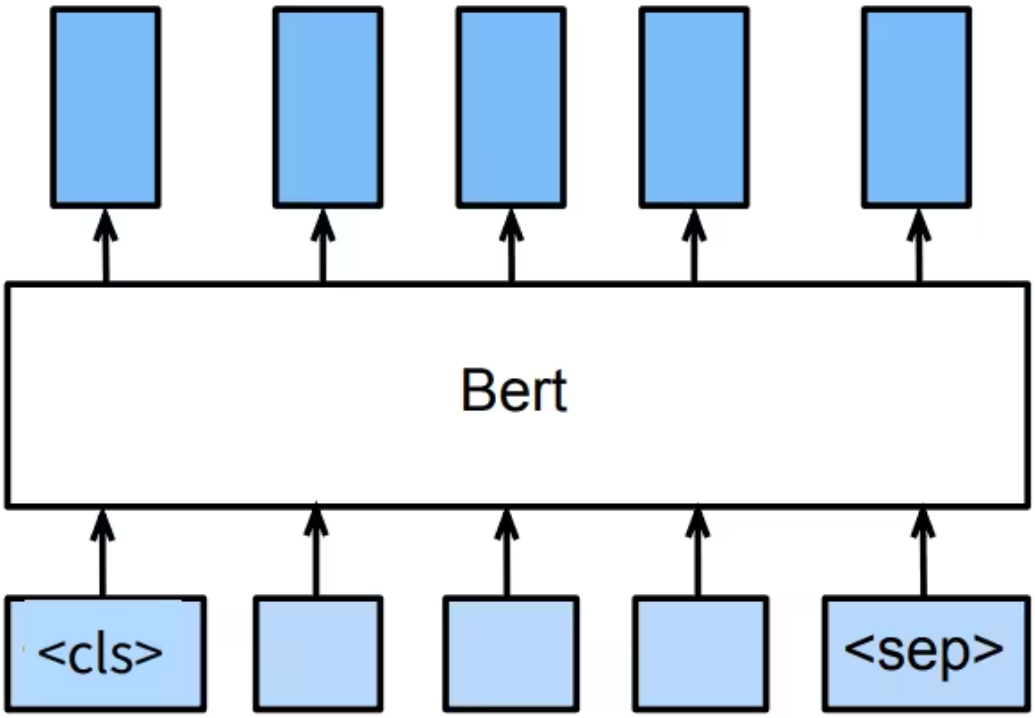
- 原数据集包含目标数据集 label 的情况，可以使用预训练好模型的任务层中对应标号的向量来初始化新模型的任务层（不常用）
- 比如用某个大型图片数据集预训练了分类模型，目标数据集都是车辆，可以用预训练模型中车辆部分图片的向量做新模型分类器的初始化，注意要对应

3.**固定一些层**

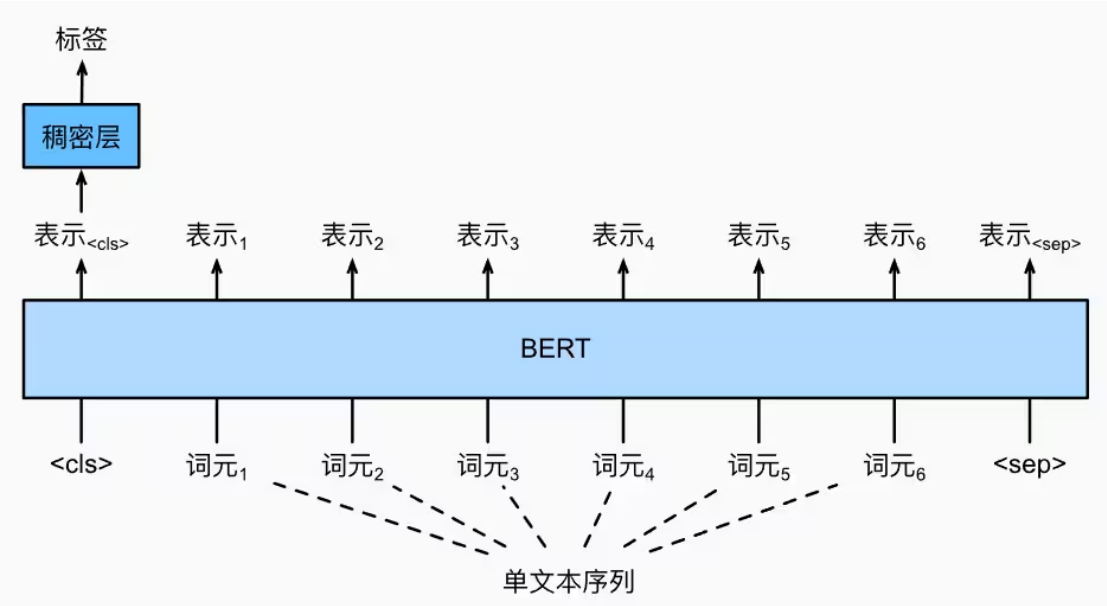
- 神经网络一般前面的层识别比较底层的特征，更加通用；后面的层更加语义化，和数据更相关
- 所以可以固定低层参数调整高层参数，以减少训练开销

微调 BERT

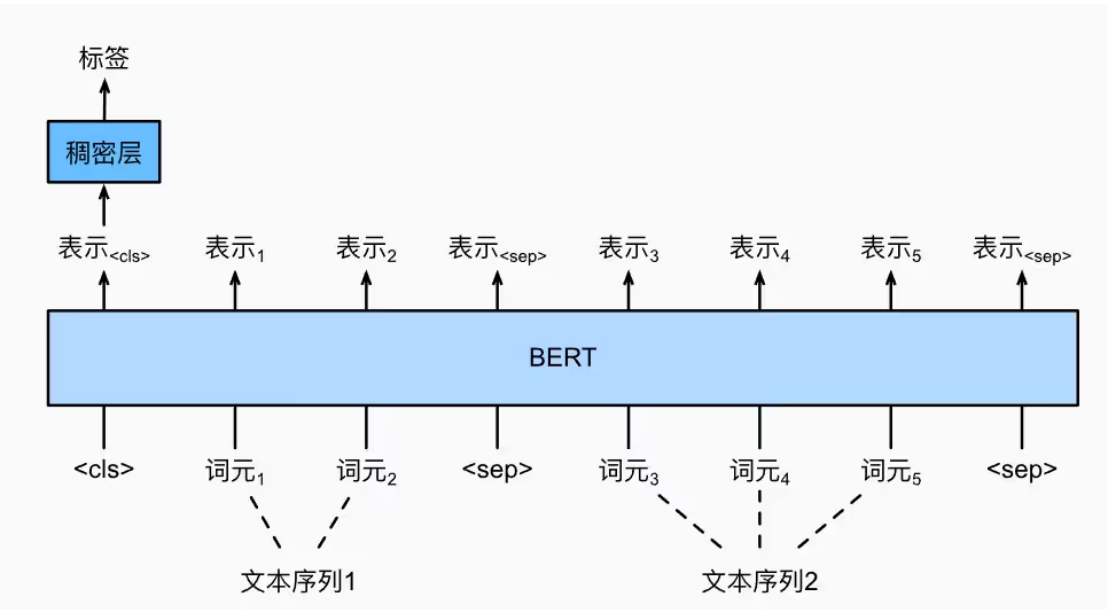
- 自然语言处理应用可以分为序列级和词元级：
  - 序列级包括单文本分类任务和文本对分类（或回归）任务
  - 词元级包括文本标注和回答
- BERT 对每个词元返回抽取了上下文信息的特征向量
- 不同的任务使用不同的特性



- **句子分类**
  - 特殊分类标记 <cls> 用于序列分类
  - 特殊分类标记 <sep> 用于标记单个文本的结束或者分隔成对文本
  - 单文本分类应用中，特殊标记 <cls> 的 BERT 表示对整个输入文本序列的信息进行编码，作为单个文本的表示，它将被送入到由全连接（稠密）层组成的小多层感知机中，以输出所有离散标签值的分布

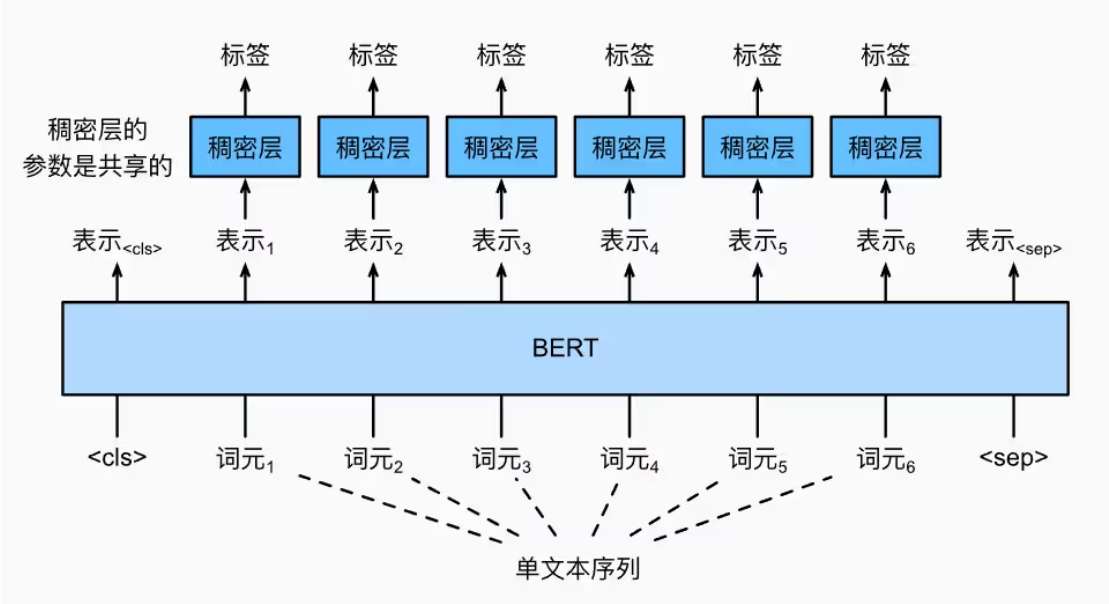


- 对于句子对也是一样的，将句子输入到 BERT 模型中，也是只将句子开始的 <cls> 标识符输出对应的特征向量，然后将这个特征向量输入到一个二分类（或者是 n 分类）输出层中做 softmax 进行分类



- **命名实体识别**
  - 识别一个词元是不是命名实体，如人名、机构、位置
  - 将非特殊词元放入全连接层分类，判断是不是命名实体
  - 句子输入到 BERT 模型之后，将非特殊词元（丢弃掉 <cls> 、<sep> 等特殊

词元，只留下真正的 token）放进全连接层进行分类，对每一个词进行词级别的分类判断（二分类或者是多分类）



- **问答**
  - 给定一个问题和描述文字，找出一个片段作为回答
  - 对片段中每个词元预测它是不是回答的开头和结束