

Lab3 Prompt Tuning 实验

1 原理概述

1.1 提示学习

提示学习 (Prompt Learning) 是自然语言处理 (NLP) 领域中的一种创新方法，特别是在利用大规模预训练语言模型 (如 BERT、GPT-3 等) 进行各种下游任务时的一种技术。它模仿人类向模型提出问题的方式，通过设计合适的“提示”——一种特定形式的输入，使得模型能更好地理解并执行任务。参考论文：The Power of Scale for Parameter-Efficient Prompt Tuning

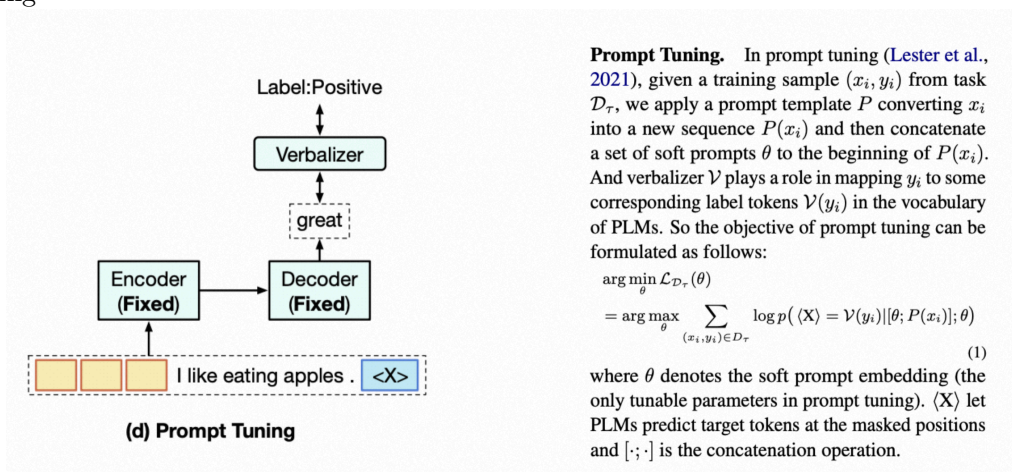


图 1: 提示学习过程

1.2 硬提示学习

硬提示学习通常涉及将预定义的标记 (通常是词或短语) 添加到模型的输入中。这些标记被称为“硬提示”，因为它们是固定不变的文本片段。这种方法简单直接，类似于提问题的形式。

例如，我们有一条评论：“电影剧情紧凑，非常精彩。”为了使用硬提示学习进行情感分析，我们可能会人为设计一个提示词或短语，比如“情感是正面的吗？”然后将这个硬提示与评论一起构成输入语句：“情感是正面的吗？电影剧情紧凑，非常精彩。”这种硬提示尝试指导模型理解其任务是判断评论的情感倾向。模型之后会根据这个包含硬提示的输入来预测情感。

原始输入： [CLS] 电影剧情紧凑，非常精彩。 [SEP]

添加硬提示： [CLS] 情感是正面的吗？ 电影剧情紧凑，非常精彩。 [SEP]

硬提示学习由于其简单性，在一些任务中可以获得不错的性能，尤其是当我们能够设计出一个非常有效的硬提示，但是这往往包含着运气成分。

1.3 软提示学习

与硬提示学习不同，软提示学习不涉及使用预设的文本片段作为输入。相反，它会在模型接收到实际输入之前加入一串可训练的嵌入向量。这些嵌入向量是模型参数的一部分，并且

会在训练过程中根据下游任务的需求进行优化和更新。因此，软提示是“软”的，因为它们是可以调整和学习的，不是固定的文本。

例如，在进行情感分析任务时，不是在“电影剧情紧凑，非常精彩。”之前加上“情感是正面的吗？”这样的硬提示，我们先得到原始输入的嵌入层输出结果，然后加上一组软提示嵌入向量，作为新的嵌入层输出结果。这些软提示嵌入向量在训练过程中自适应地学习，修改它们的值以提高模型识别正面或负面情感的能力。

$$\begin{aligned} \text{原始输入: } & [CLS] \text{ 电影剧情紧凑, 非常精彩. } [SEP] \\ \text{原始输入的嵌入表示: } & \vec{x}_{cls}, \vec{x}_1, \vec{x}_2, \dots, \vec{x}_m, \vec{x}_{sep} \\ \text{添加软提示嵌入: } & \overbrace{\vec{e}_1, \vec{e}_2, \vec{e}_3, \dots, \vec{e}_n}^{\text{Soft Prompt}}, \vec{x}_{cls}, \vec{x}_1, \vec{x}_2, \dots, \vec{x}_m, \vec{x}_{sep} \end{aligned}$$

在实际应用中，软提示通常表现出更大的灵活性和效率，因为它们允许模型在不同任务和领域间灵活转换，并在固定模型参数的情况下进行微调。

2 实验内容

2.1 阅读硬提示代码

- https://github.com/mindspore-courses/step_into_llm/blob/master/Season1.step_into_chatgpt/7.Prompt/mindprompt_emotect.ipynb
- 阅读代码，了解如何实现硬提示调优

2.2 实现软提示训练

- 以提供的硬提示学习框架为基础，实现软提示训练，进行了以下修改
- 数据预处理：
 - 目标任务：情感分类。
 - 数据集文件：可在 `prompt tuning/data` 文件夹中找到，包含 `rt - polarity.neg` 和 `rt - polarity.pos` 两部分。
 - 预处理：请参考 `prompt tuning/TextCNN_Mindspore.ipynb` 中的代码进行数据预处理。
- 添加软提示：将框架中的自然语言提示模板替换为一系列可学习的嵌入向量“软提示”
- 进行训练：只设置软提示的嵌入向量为可训练参数，利用预处理好的数据进行训练
- 测试评估：在训练完成后，设计一些样例进行测试，以便评定模型的效果。