NLP方向总结-提示学习
定义
传统监督学习
训练一个模型来接受输入x并预测输出y的P(y|x)
LM适应于下游任务
基于（prompting）提示的学习
基于语言模型
将下游任务适应于LM
原始输入x被使用模板修改成一个文本字符串提示x'，其中有一些未填充的槽，然后语言模型被用来概率性地填充未填充的信息，得到一个最终的字符串x^，从中可以得出最终的输出y
例如，当识别一个社交媒体帖子的情感时，"我今天错过了公交车。"，我们可以添加提示 "我感觉如此___"，并要求LM用一个带有情感的词来填补空白。或者如果我们选择提示 "英语。I missed the bus today. 法语：")，LM可能会用法语翻译来填空。
其它不同任务的表述形式见表3
已有模型
https://github.com/clue-ai/PromptCLUE
https://github.com/IDEA-CCNL/Fengshenbang-LM/tree/main/fengshen/examples/unimc
结构：图1总结拓扑图
预训练模型
表13进行了总结
缩写
SLM：标准语言模型
自回归LM
例如GPT
CTR：损坏的文本重建
对输入句子的噪声部分计算损失，将处理后的文本恢复到未损坏的状态
例如BERT
FTR：全文重构
通过计算整个输入文本的损失来重构文本
例如BART等翻译模型
L2R
从左到右的预训练方法
自回归式LM的一种，它预测下一个单词
Mask
Mask的方式
目的是根据包围的上下文预测masked的文本片段
适用于cloze类型的提示任务
适用于文本分类、自然语言推理，和提取式问答
PLM
Prefix LM：前缀语言模型
前缀LM是一种从左到右的LM，它以前缀序列x为条件对y进行解码，该序列由相同的模型参数编码，但有一个全连接的mask
En-De
Encoder-decoder结构
提示工程
是创建一个提示函数fprompt(x)的过程，该函数能在下游任务中产生最有效的表现
提示形式
填空提示
填补文本字符串的空白
前缀提示
延续字符串的前缀
有关生成的任务
多输入的任务
如文本对分类，提示模板必须包含两个输入的空间，[X1]和[X2]
提示模板
手动
自动模板学习
离散方法(硬提示)-寻找合适的词汇
Prompt Mining
基于挖掘的方法，在给定一组训练输入x和输出y的情况下，自动寻找模板
提示语转述
采用现有的种子提示语（例如手工构建的或挖掘的），并将其转述为一组其他的候选提示语，然后选择在目标任务上达到最高训练精度的提示语
基于梯度的搜索
实际的token上寻找能够触发底层预训练的LM以产生所需的目标预测的短序列
提示语生成
将提示语的生成视为文本生成任务，并使用标准的自然语言生成模型来执行这一任务
提示评分
使用单向LM对这些填充的提示进行打分，选择具有最高LM概率的一个
连续方法(软提示)-寻找嵌入
构建提示的目的是找到一种能让LM有效执行任务的方法，而不是供人使用，所以没有必要将提示限定为人类可理解的自然语言
(1) 放宽模板词嵌入是自然语言（如英语）词嵌入的限制。(2)取消了模板由预训练的LM的参数决定的限制
前缀优化
将连续的特定任务向量序列预加到输入中的方法，同时保持LM参数冻结
用离散提示语初始化的优化
用已经创建的提示语或用离散提示语搜索方法发现的提示语来初始化连续提示语的搜索
软硬提示混合优化
答案工程
寻找答案空间Z和与原始输出Y的映射
答案形式
Token
跨度
句子
答案空间设计
离散答案
答案解析
这些方法从初始答案空间Z'开始，然后使用解析来扩展这个答案空间，以扩大其覆盖范围
先修剪再搜索
首先产生一个由几个可信的答案Z组成的初始修剪过的答案空间，然后一个算法在这个修剪过的空间上进一步搜索，选择最终的答案集
标签分解
用于关系抽取
连续答案
多提示学习
多个提示提升性能（图4）：（1）利用不同提示的互补优势，（2）减轻提示工程的成本，因为选择一个表现最好的提示是具有挑战性的，（3）稳定下游任务的性能。
基于提示的训练策略
参数更新
（i）是否优化基础LM的参数，（ii）是否有额外的提示相关参数，（iii）如果有额外的提示相关参数，这些参数是否被微调。
论文
P-Tuning v2： Prompt Tuning Can Be Comparable to Fine-tuning Universally Across Scales and Tasks
P-Tuning v2: 在不同的规模和任务中，提示优化可以与微调相提并论
定义
代码：https://github.com/THUDM/P-tuning-v2
微调模型的缺点
为目标任务更新整个模型参数集，训练过程中很耗费内存，因为必须存储所有参数的梯度和优化器状态，推理过程中为每个任务保留一份模型参数
离散提示法
冻结了预训练好的模型的所有参数，并使用一个自然的语言提示来查询语言模型，导致次优的性能
Prompt tuning：提示优化（图2）
但当模型规模不大，特别是小于100亿个参数时，它仍然不如微调的表现，只有当使用大模型时，才会出现“涌现”能力
缺点
缺少跨规模的普遍性
在简单的分类任务上，当模型规模超过100亿个参数时，提示优化可以与微调相媲美。然而，对于被广泛使用的中型模型（从100M到1B），提示优化的表现比微调差很多。
缺少跨任务的普遍性
在一些NLU基准上显示了优势，但提示优化在困难序列标注任务上的有效性并没有得到验证。而且与verbalizers不兼容
提示优化（P-Tuning, Prompt tuning）
提示学习策略中，冻结预训练模型参数，去优化提示的方式。因为提示学习的好坏与提示词（或本文是提示嵌入）的好坏呈正相关。
离散提示：即人工设计的提示
例如判断一个句子的情感: “很不错的电影”，设计提示: 它是[MASK]，输入变成: 它是[MASK]，很不错的电影，，预测[MASK]的单词，例如结果“好”或“坏”，即可知道情感是正向还是负向
嵌入提示：即嵌入的提示
还是判断一个句子的情感: “很不错的电影”，提示变成了连续的嵌入向量: 那么整个输入也是一个向量，[提示的嵌入向量] + [MASK]+ [很不错的电影的嵌入向量]。 那么我们优化提示的嵌入向量即可。
微调VS提示学习
1. 微调需要在训练时存储所有参数的梯度和优化器状态，需要更大显存
2. 微调的每个任务都需要有单独的模型参数
3. P-tuning v1超过100亿个参数时提示学习和微调性能才能相等
4. 提示学习序列标注任务中效果不好
提示学习的问题
1. 可调控的参数量有限，大多数语言模型目前只能支持512的最大序列长度（由于注意力的二次计算复杂性的成本），所以填充的提示长度也是有限的。
模型（图2b）
P-tuning v2
深度提示优化，改进来自于对预训练模型的每一层应用连续的提示，而不是单纯的输入层，缩小了在各种环境下进行微调的差距，特别是对于小模型和困难任务
与微调所有参数相比，P-tuning v2每个任务的可训练参数为0.1%到3%
对比以前的其它提示优化方法的不同
不同的NLU任务通常会在不同的提示长度
多任务学习在对单个任务进行微调之前
标注之上应用了一个随机初始化的分类头，代替先前的语言模型头来预测verbalizers，因为verbalizers与序列标注不兼容
注意点：只在第17-24层添加提示语可以产生与所有层非常接近的性能，这说明部分核心层进行优化就可以了
结构（图2）
1. 通过前缀的方式添加
2. 在transformer结构的每层的前面添加连续的嵌入
对比只在嵌入层加提示，这样训练优化更平稳
添加接近输出层的提示比浅层提示效果更好
3. 对比第一代提示优化，删除了带有LM头的verbalizers
使用CLS进行分类
4. 其它
1.连续提示的参数重新初始化对性影响
2. 不同的提示长度对性能影响
任务越难，提示越长更好
3. 多任务学习对性能影响
Zero-Shot Learners for Natural Language Understanding via a Unified Multiple Choice Perspective
统一的多选视角实现自然语言理解的zero-shot学习
定义
代码：https://github.com/IDEA-CCNL/Fengshenbang-LM/tree/main/fengshen/examples/unimc
优点
2.35亿个参数
一般的提示学习需要在大模型上运行，GPT-3有175B，FLAN有137B，PaLM有540B
提示学习，兼容多个任务兼容，如文本分类、常识推理、核心推理和情感分析
zero-shot泛化性好
支持多选，即多标签
参数更新只发生在MC训练阶段
减少人工处理，只制定了一个候选选项提示格式和一个问题提示格式
UniMC：统一多选模型
将有问题的分类器转换为选项
一个选项mask token[O-MASK]来预测每个选项前的 "是 "或 "否"
进行选项MLM（O-MLM）来恢复每个选项的 "是 "或 "否"
MC： Multiple Choice
PLMs：预训练语言模型
对比其他模型：GPT2，GPT3，T0，FLAN，PaLM，GaLM，UnifiedQA
实验
使用ALBERT-xxlargeV2模型作为主干
token长度512
训练一个1 epoch
将每个任务的样本数设置为20K，目的是防止模型被特定任务所支配
使用14个数据集，表8
2.35亿个参数
任务类型：自然语言推理，情感分析，分类
模型
多种任务统一输入（图3）
一个多选问题通常由三个部分组成，包括选项、问题和段落
主干模型：Albert或Roberta
每个选项前加 [O-MASK]， 代表答案的"是 "或 "否"
使用Mask的方式确保选项之间不相互影响，图5示意图
常规MLM任务
选项mask语言模型（O-MLM）
我们Mask [O-MASK] token来预测 "是 "或 "不是"
选项预测（OP）方法
为了学习选项之间的互斥特性，计算一个具有ground truth标签分布Y的交叉熵损失，公式（4）
GPT Understands, Too
P-tuning v1：GPT的NLU任务
定义
GPT模型对于NLU任务
使用P-tuning--即采用可训练连续提示嵌入，GPT在NLU任务上可以比类似规模的BERT更好或更有可比性
打破了GPT只能生成而不能理解的刻板印象，其实ChatGPT已经表现出来这个能力了，但本论文是2021年发布的，ChatGPTshi 2022年底发布的
P-tuning
P-tuning利用少数连续的自由参数作为提示语输入到预训练语言模型中。然后，使用梯度下降法对连续提示进行优化，作为离散提示搜索的一种替代。
P-tuning用其差异化的输出嵌入取代了预训练语言模型的输入嵌入
大模型的问题
下游任务的微调对那些万亿规模的模型几乎不起作用。即使是对于许多次的微调设置，这些模型仍然太大，无法快速记忆微调样本
如何使用提示学习
GPT-3和DALL-E利用人工筛选的提示来引导模型的下游应用，然而，手工提示搜索严重依赖不切实际的大型验证集，但其性能也很不稳定
模型（图2）
离散提示--如何变成更好的连续连续提示，所以训练嵌入张量
挑战
初始化
原始词嵌入e经过预训练已经变得高度离散，连续提示的向量hi不能随机初始化，否则优化器将很容易陷入局部最小值
关联性
嵌入hi中的每个嵌入向量都是有关联的
使用LSTM头解决离关联性和初始化