**Chatglm2-6B微调方案简介（JamePeng）**

前提：本次比赛的方案可以有很多种，之所以选择微调模型来增强，还是希望通过比赛过程中更多的让模型加入实战中。利用模型本身对文字的理解来处理问题。前期设置了loss在0.02以下的合格目标

优势：如果全量训练三类模型的话，时间和计算量肯定不够，使用微调训练的方式，只需要加载训练权重就可以增强Chatglm2-6B某方面的能力（但会丧失基础语言交流），不需要的时候就卸载模型重新加载即可（恢复正常的语言交流能力）。

使用Chatglm\_Ptuning这个类，通过传入不同类型的Ptuning枚举类来自动加载不同任务类型的微调权重（使用NOTHING的话则默认加载原始模型）

*# 创建一个枚举类型, 不同ptuning分类*

*class PtuningType(Enum):*

*Nothing = 0*

*Classify = 1*

*Keywords = 2*

*NL2SQL = 3*



使用完成后使用unload\_model函数来释放已经加载的权重

*# 卸载掉已经装在权重的模型*

*def unload\_model(self):*

*del self.model*

*del self.tokenizer*

*torch.cuda.empty\_cache()*

一、 微调一个question关键词提取器Keywords，将5000个问题和可能会出现的统计题，抽出600道各类型的题目做数据化标注。 【训练长度为：256】

根据程序的需要输出提取到核心关键词，根据关键词可以方便队长的算法进行检索该关键词问题的处理方式。

用ptuning对chatglm2-6b进行微调训练，第一版使用A榜的题目进行抽取大体的题目类型300道题进行训练，step从100到200，loss趋于稳定在0.017-0.024的合格区间。

测试后确认这个prompt能够让chatglm比较好的提取关键词的任务

*role\_prompt = '''*

*请帮我从以下句子中提取关键词。这些关键词是句子中最重要、最能概括句子主题的词汇。通过这些关键词，你可以更好地理解句子的内容。你只需要回答文本中的关键词,不要回答其他内容.*

*用户输入：*

*'''*

0823版本的keyword提取器的训练loss达到0.022合格的分类水平，5000个问题的关键词提取效果合格（不同公司名称、简称和地名会影响提取，核心关键词提取优良）。

0903版本的训练集扩充到4000多道题，step提升到250，训练集根据问题分类来提取：

type1和type2的题目只提取核心关键词（不含年份和公司名）

type3的题目提取核心提问句

训练loss达到0.015较好的分类水平，AB榜的问题提取关键词有较好的提取关键词效果

二、编写一个question分类器Classify，设计一个prompt（问题模版，选择题模板）来让AI进行我们想要达到的分类，将5000个问题和可能会出现的统计题抽出600道各类型的题目做数据化标注。【训练长度为：512】

（0）chatglm对选择题有一点的理解能力，默认无微调的状态也有一定的分类能力，比如A选项公司信息，B选项员工信息的分类表现非常不错，但是对A榜开始的计算题，B榜开始的统计题就难以和单纯的财务信息检索进行区分。希望通过微调来增强他的分类能力

（1）最早期的一次微调，一开始的效果比较糟糕，最后发现是训练字段小于传入的模版+问题的字符串，导致训练过程中的传入字段丢失截断。loss一直在0.15下不去，实用情况比较糟糕。

（2）通过统计问题和答案的长度，最后调整了max\_source\_length参数符合max（问题+答案)的条件，确认了max\_source\_length为512，max\_target\_length为128，使用ptuning对chatglm2-6b进行微调训练，step为250

（3）为了获取更好的效果，尝试对prompt不断进行调整，从一开始的只有选项的题目，到有一定信息的选项描述（高频的关键字等），对微调训练有很大的帮助，loss的下降比较明显。

*def \_get\_classify\_prompt(self, question) -> str:*

*classify\_prompt = '''*

*请问“{}”是属于下面哪个类别的问题?*

*A: 公司基本信息,包含股票简称, 公司名称, 外文名称, 法定代表人, 注册地址, 办公地址, 公司网址网站, 电子信箱等.*

*B: 公司员工信息,包含员工人数, 员工专业, 员工类别, 员工教育程度等.*

*C: 财务报表相关内容, 包含资产负债表, 现金流量表, 利润表 中存在的字段, 包括费用, 资产，金额，收入等.*

*D: 计算题,无法从年报中直接获得,需要根据计算公式获得, 包括增长率, 率, 比率, 比重,占比等.*

*E: 统计题，需要从题目获取检索条件，在数据集/数据库中进行检索、过滤、排序后获得结果.*

*F: 开放性问题,包括介绍情况,介绍方法,分析情况,分析影响,什么是XXX.*

*你只需要回答字母编号, 不要回答字母编号及选项文本外的其他内容.*

*'''.format(question)*

*return classify\_prompt*

0826版本的question分类器的训练loss达到0.014（最优0.005）优秀的分类水平，5000个问题加统计题的分类效果良好。

目前0829版本修复一些语言描述模糊的分类【详细信息，详细数据等描述】，loss最优达到0.003（平均0.006），提升分类的准确度

三、根据统计题的需求，配合队长的提取表格，使用column0的字段将数据入库SQLite3。

设计一个NL2SQL能力的微调模型，希望是通过已知的数据库字段，给出一个模版和问题的字符串提问模型，可以获取问题方向可用于数据库检索的SQL语句.【训练长度为：2048】

(0)早期尝试用公共的NL2SQL训练集（2019年NL2SQL大赛的公开标注数据集），大概40000多道题目，希望增强Chatglm2-6B的SQL能力。结果发现，不仅没有能够很好对题目进行SQL生成，而且丧失了基本的对话能力，自认为是题目里出现过的张三。

(0-1)再次尝试公共训练集，这次加上了比赛题目的题型，希望强化SQL后可以对比赛题目也能有比较好的表现。训练后发现，小规模的标注60道题难以对原来较大规模的题目，而且发现原来的训练集其实也是有很多问题，最后放弃掉公共的NL2SQL的训练集。

(0-2)根据前面两个尝试，最后决定自己来标注训练集，为了加快效率，使用chatgpt来协助训练集的生成，第一轮扩充到了200多道题目

(1)设计提问的模版，带催眠、带数据库字段、要求不能使用字段外的回答、给出一些样例模版。多种方式来引导AI按照我们想要的方向来回答问题。

测试过：

<1>只要求他根据问题来给出SQL，chatglm2-6b的数据库能力比较糟糕，生成的SQL语句几乎无法参考使用。

<2>给出数据库字段，要求他根据问题来生成SQL语句，有一点提升，但生成的SQL语句格式依旧无法参考使用，关键词的提取有一点强化。

<3>催眠它是SQL工程师，可以专业地解决NL2SQL的问题，并且在第二点的基础上，加上表名的表述，还有两个问题和输出的模版示例，可以获得格式和字段比较合理的SQL语句。因此有比较大的提升空间。

(2)根据第三点的测试，构建针对其模版格式的问题训练集，根据比赛提供的四个模版扩充了24种问法，并且强化了除了查询以外的SQL语句，如排序，输出范围，计数，求和，单字段检索，多字段检索，多字段检索多字段，字段的过滤等，不同比较级的语言描述（高低，大小，多少，否定词+理想，中文数字和阿拉伯数字），扩充到了200多个高质量的经过个人核验校对的的训练数据集。

(3)数据集的问题长度统计，max(question)为1980，按照合理的训练长度设计为2048，输出的语句max(query)为249，设计target长度为256. 进行200轮的ptuning微调训练，最终结果获得loss0.005（最优0.0034），evaluate评测数据的稳定度达到99.56，

0829版本，修正部分字段描述的差异，【成本与总成本，收入与总收入，非流动资产与非流动金融资产等描述】，可以更加准确的提取字段应对比赛的统计类问题。

0903版本，由【Steven宋】继续优化：

1）调整agent和example的模版，加入了统计题的问答模版，同时强化对数字的转化认知（第一版发现Chatglm2对长数字的认知能力比较糟糕，阿拉伯数字存在多和少数字，中文数字表达的数字转换很糟糕，透过prompt的形式告诉他数字的长度和转换来增强）

2） --max\_source\_length 2200 --max\_target\_length 300 参数进行了一定的调整，为了适应

3）扩充NL2SQL训练集，从400道题扩充到近1100道题，

4）训练Step增加到了600 step，loss最后在0.004-0.005之间波动，呈现较好的loss和稳定性