

## 赛道二-xiaoerbaode-代码方案

### 一、数据处理部分

1、首先运行 count\_difficulty.py

```
python count_difficulty.py
```

该代码生成一个 p\_0.npy 文件，位于 data 目录下，包含题目平均得分（难度）低于 0.75 的题目 id

2、然后运行 data\_fenxi.py

```
python data_fenxi.py
```

该代码进行数据预处理，生成以下文件，位于 data 目录下：

all\_data.json 将 problem\_act\_train.json、problem\_act\_train2.json、problem\_act\_test\_new.json 整合在一起后的数据，按学生 id 排序

train\_data.npy 过滤答题序列长度小于 5 的数据，该文件中所有答题序列不包含线上测试集里面的答题记录

test\_data.npy 过滤答题序列长度小于 5 的数据，该文件中所有答题序列均包含线上测试集里面的答题记录

student2id 将学生 id 转为字典

problem2id 将题目 id 转为字典

problem2kpoint 讲题目 id 转为知识点 id 字典

3、最后运行 generate\_data.py

```
python generate_data.py
```

该代码生成训练数据集、验证数据集和线上测试数据集，其中训练/验证采取十折交叉，生成文件位于 data/data\_30 目录下。我们过滤了难度值低于 0.75 的题目，生成线下数据时前 60% 数据进行模型训练，后 40% 数据进行模型验证，线上数据用已知答案的答题数据用于建模。

### 二、模型训练部分

该部分代码在 ckt\_model 文件夹下，运行完第一部分三个代码后，进入 ckt\_model 文件夹，运行以下代码：

```
python run_all.py
```

```
python test_all.py runs
```

运行完成后，result 文件夹下生成十个预测文件

### 三、预测部分

combine.py 用于单个文件预测

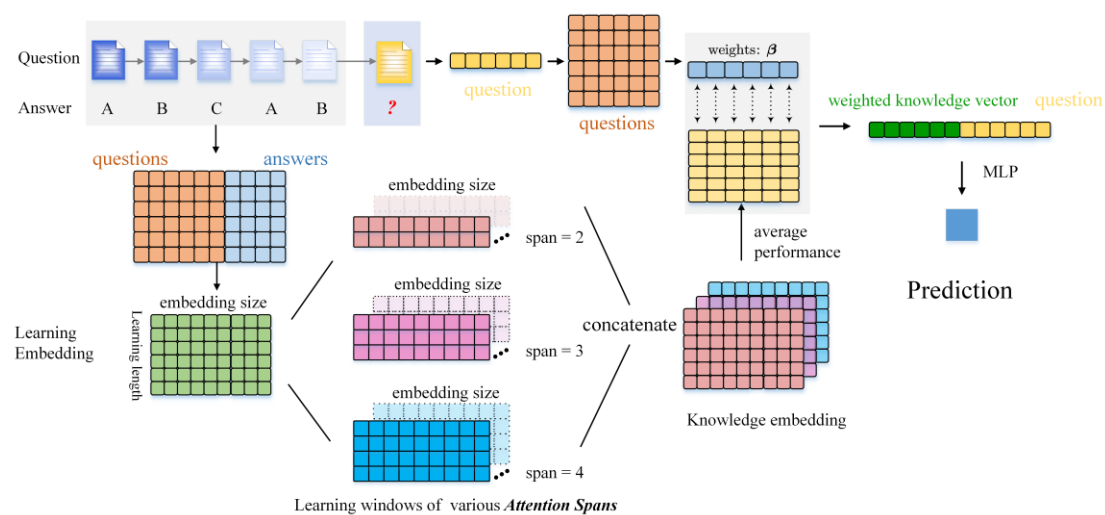
combine\_v2.py 用于多个文件融合预测，融合策略是取多个预测的平均值

预测完成后在当前目录生成 submission.json 文件，即为最终预测结果

### 四、运行说明

Python 版本：python3.7

Tensorflow 版本：tensorflow-gpu==2.3.0



模型示意图