1.代码组织结构介绍:

```
|-- work
        |-- row_data
            |-- BQ
            |-- LCQMC
            |-- OPPO
            |-- train.txt(脚本会自动生成)
            |-- dev.txt(脚本会自动生成)
            |-- test_B_1118.tsv
        |-- user_data
            |--eda_data
                |-- chinese-words.txt
                |-- cilin.txt
                |-- same_pinyin.txt
                |-- gaiic_track3_round1_train_20210220.tsv
                |-- word_0.2pinyin_0.15.txt
                |-- word_data.txt
                |-- pinyin_data.txt
                |-- gaiic_train_eda.txt
            |-- gaiic_eda_real_pool_gru_droo0.4+0.2_checkpoints
                |-- model_17800_acc_0.8805
            |-- stop_data
                |-- test_B.txt
                |-- stopword
                |-- stopword_2
            |--tmp_result
         |--code
            |-- train.py
            |-- GRU_pool_model.py
            |-- data.py
            |-- predict.py
            |-- eda.py
            |-- dd_lac_process.py
            |-- tokenization.py
            |-- tool.py
            |-- result_deal.py
```

代码功能如下表

code name	功能
train.py	训练、验证、保存模型 完成3个epoch的训练并输出验证结果 保存模型文件到指定文件夹
GRU_pool_model.py	模型文件，构建网络结构，对输入的向量进行特征分析，得到分类结果
data.py	构建数据，将数据采样为batch-size大小
predict.py	预测代码，完成测试集的数据构建和预测结果
eda.py	数据增强、同音同调、

	同音异调和替换同义词近义词
dd_lac_process.py	构建词性、句子依存关系特征
tokenization.py	提取词性、句子依存关系所需要的分词
tool.py	封装数据处理所需要的函数
result_deal.py	数据处理
dev_log.py	运行日志保存 运行 main_test.sh 模型预测验证集效果，最终精度日志将保存在work/log/dev_predcit.log中

user_data介绍:

Name	内容
chinese-words.txt	字频表
cilin.txt	同义词词林
same_pinyin.txt	同音字词典
gaiic_track3_round1_train_20210220.tsv	文本匹配数据集
word_0.2pinyin_0.15.txt	增强数据集
word_data.txt	同义词近义词替换文本
pinyin_data.txt	同音同调、同音异调替换文本
gaiic_train_eda.txt	增强后的数据集

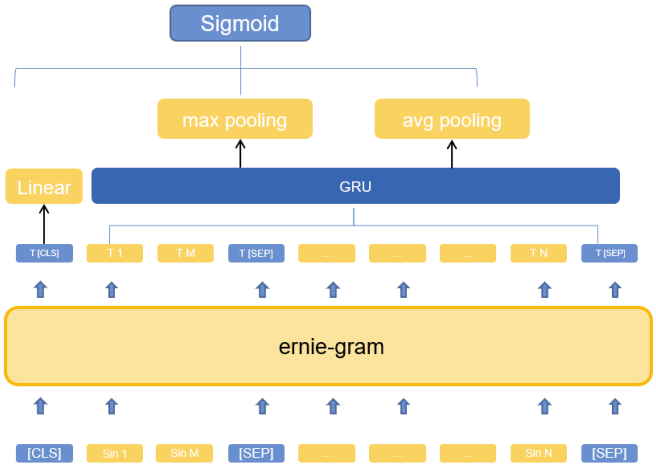
2.数据增强/清洗策略

数据增强策略使用同音同调、同音异调和替换同义词近义词策略，同义词近义词替换比例为20%，同音同调、同音异调替换比例为15%，替换后的文本随机选择50%进行交换，增强数据健壮性。详情请见eda.py。

3.模型设计和优化

模型选取ERINE-Gram为预训练模型，在模型后对预训练输出分别接线性层和GRU网络，再使用最大池化和平均池化对特征进行提炼，获得更优化的特征，最后再拼接起来进行分类。具体结构如下图。详情请见GRU_pool_model.py。

在网络结构方面通过调整参数等方法，在ERINE-Gram后提高drop_out比例至0.2，在分类输出前继续使用drop_out防止模型过拟合，增强模型鲁棒性。



4..训练脚本/代码:main_trian.sh 训练时长3.5h左右

如需运行请在终端输入: sh ./work/main_train.sh

5..测试脚本/代码:main_test.sh.　预测时长20min左右
请最后在work/prediction_result路径查看
如需运行请在终端输入:sh ./work/main_test.sh

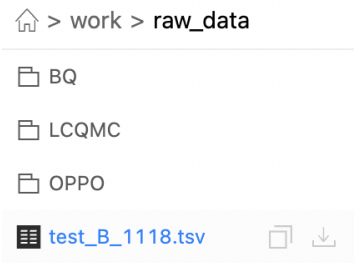
6.参考链接：以2019年数据智能创新应用大赛——基于Adversarial Attack的问题等价性判别比赛为背景https://github.com/activemodest/DIAC

gaiic2021-track3-小布助手对话短文本语义匹配复赛rank3、决赛rank4
https://github.com/daniellibin/gaiic2021_track3_querySim

百度LAC词性开源：https://github.com/baidu/lac

百度DDParser依存关系开源：https://github.com/baidu/DDParser

【注意】1.因为ai studio公开版本无法加载1G以上的文件，所以在脚本文件里写了依赖下载命令，直接运行脚本即可。
2.复现更换raw_data文件时，请按照下图方式更换。



若在复现过程中若遇到任何问题，可电话联系：18915567597，感谢！