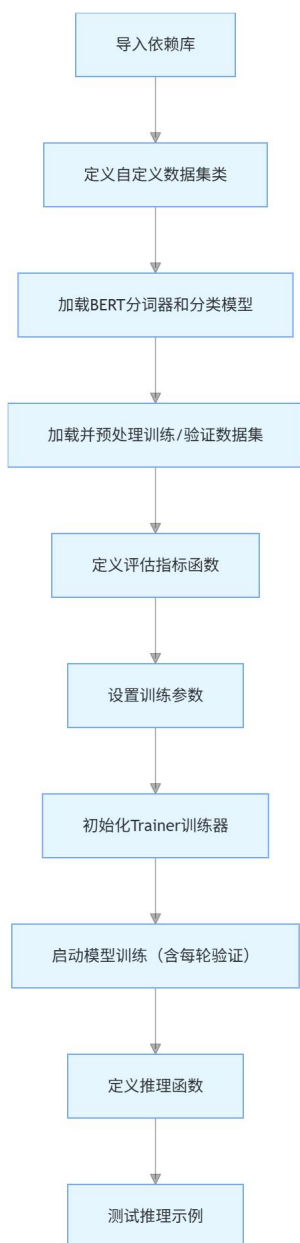


# 山东大学计算机科学与技术学院

## 大数据分析实践课程实验报告

学号：202300130098	姓名：马浩鑫	班级：23 级数据
实验题目：BERT 配置及实践		
实验学时：2 课时	实验日期：	
<p>实验目标：</p> <p>对动手实践利用机器学习方法分析大规模数据有进一步了解，并学习如何利用远程环境进行工程代码的调试.</p>		
<p>实验描述：</p> <p>配置环境：</p> <p>安装 python、pycharm 及必要相关库、cuda 等</p> <p>熟悉 PyTorch 框架下，利用预训练的 transformers 的预训练 BERT 模型对 MRPC 数据集进行同义预测的 pipeline. 尝试理解数据是如何预处理，模型是怎么读入数据，是如何进行推理，如何进行评价的.</p> <p>(代码逻辑：对 BERT 进行微调，每个句子对用 BERT 指定分隔符 [SEP] 连接后，通过 BERT 得到合成句子的 representation. 再通过通过一个两层的多层感知机得到分类结果. 这里预训练 BERT 模型使用的是 HuggingFace 的 BERT-base-uncased)</p> <p>数据集</p> <p>MRPC (Microsoft Research Paraphrase Corpus) 包含了 5800 个句子对，有的是同义的，有的是不同义的，是否同义由一个二元标签进行描述.</p> <p>代码流程：</p>		



完成相关导入和定义后，加载 MRPC 数据集清洗并导入 BERT 模型进行与训练，在每轮训练时产出评分（分为准确率、f1 评分、准确率和召回率），最终使用训练好的数据对给出的两组句子对进行判别是否语义一致。相关得分输出如下：

```
{'eval_loss': 0.6553844213485718, 'eval_accuracy': 0.8411042944785276, 'eval_f1': 0.8840125391849529, 'eval_precision': 0.8575152041702867, 'eval_recall': 0.9121996303142329, 'eval_runtime': 7.8564, 'eval_samples_per_second': 207.474, 'eval_steps_per_second': 25.966, 'epoch': 3.0}
{'train_runtime': 194.5773, 'train_samples_per_second': 60.392, 'train_steps_per_second': 7.555, 'train_loss': 0.33956758667822595, 'epoch': 3.0}
```

在仅训练三轮的情况下准确率有 84%，性能还是比较好的。

最终对两组句子对鉴别如下：

```
print(predict("A man is playing guitar.", "Someone is playing a guitar."))
```

```
print(predict("The cat sits on the mat.", "A dog chases a ball."))
```

测试推理：

一致

不一致

结论分析：  
数据处理有效性：自定义 MRPCStructuredDataset 类完成了 TSV 格式数据的清洗（空值过滤、无效标签筛选）、BERT 适配的句子对编码（截断 / 填充至 128 长度），确保输入数据符合模型要求，无格式错误导致的训练中断风险；

评估体系完整性：集成了准确率（Accuracy）、精确率（Precision）、召回率（Recall）、F1 值四个核心指标，覆盖二分类任务的关键评估维度，可全面反映模型性能；

关键指标

指标	意义与实验结论
准确率	反映模型整体判断正确的比例，若准确率低于 80%，需排查数据加载（如标签是否反转）或训练轮数不足问题；
F1 值	MRPC 任务的核心指标（因数据存在一定类别不平衡），F1 值越高说明模型在“一致 / 不一致”分类上的综合性能越好；
精确率 / 召回率	若精确率远高于召回率，说明模型“漏判”较多（把实际一致的句子判为不一致）；若召回率远高于精确率，说明模型“误判”较多（把不一致的句子判为一致）；