|  |  |
| --- | --- |
| **自然语言处理及应用**  **实验报告** | |
|  | |
|  |  |
| **名称** | 大作业\_sBert (group 4) |
| **姓名** | 卢佳源 |
| **班级** | 人工智能91 |
| **学号** | 2191121196 |
| Email | bjlujiayuan@126.com |
| **日期** | 2021-11-6 |

# 实验目的

基于Sentence-Bert模型进行文本查重，包括检验sBert模型对文章查重任务的适用性、构建面向查重的数据库，以及对查重应用进行前端设计。

# 实验环境

OS：Linux操作系统的ubuntu20.04版本

IDE：VScode,anaconda3(‘base’:conda)

Terminal：gnome-terminal -t $TITLE -x

Language：python(3.8.8 64-bit)

Tools：PyQt5（前端）

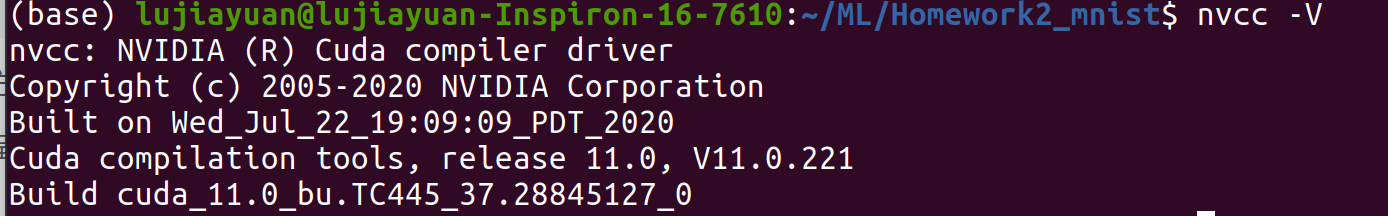
PATH：

前端：/home/lujiayuan/nlp/Big\_Homework/NLPLarge

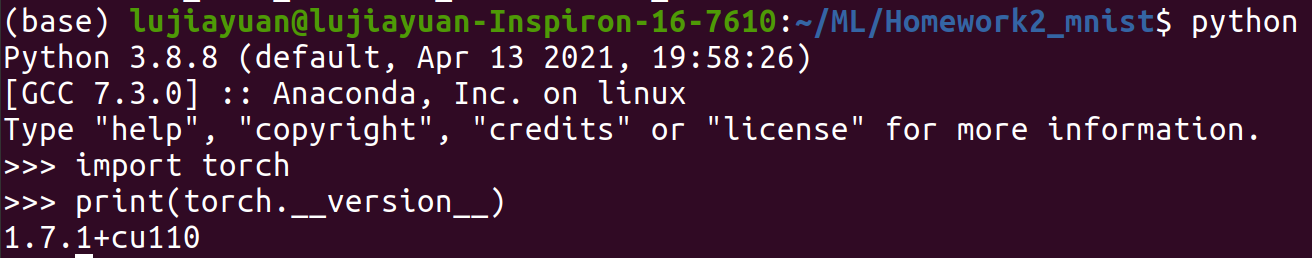
后端：/home/lujiayuan/nlp/Big\_Homework/split\_sentence\_new.py & test1.py & TranslateTool.py & utils.py & evaluate.py & fine\_tune.py & buildDatabase.py & config.py

Data:/home/lujiayuan/nlp/Big\_Homework/lcqmc & government & govern\_pos\_neg &数据集1

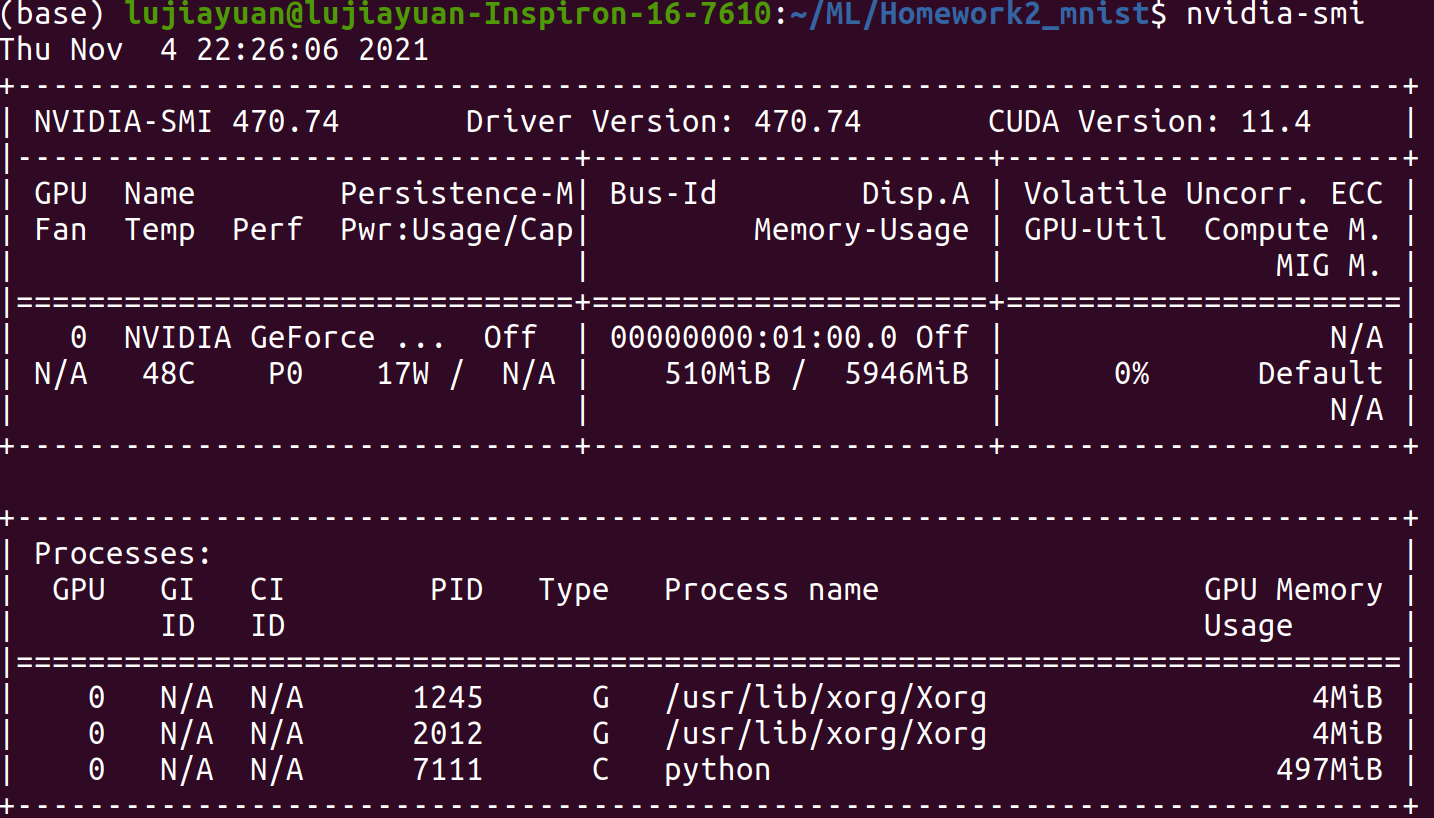
CUDA版本：



pytorch版本：



显卡信息：Name: GA106M [GeForce RTX 3060 Mobile / Max-Q]

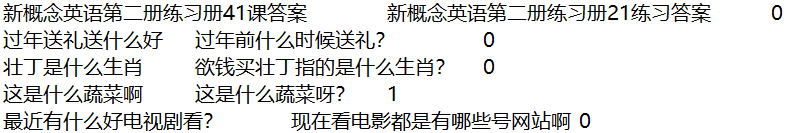


# 实验方法

1. 数据集准备：

我们准备了两个数据集，lcqmc和政府文件数据集。

* lcqmc：针对不同领域从百度问答中抽取高频相关问题，训练集大小为240000对



* 政府文件数据集：利用爬虫从新华网上找到16篇政府文件，并进行中文分句，最终得到八千多个句子。
* 自监督：（我负责构建正样本部分）由于政府文件是从网上爬下来的，分句后得到的数据集没有标定，同时还没有正负样本，因此我们需要用自监督的方式构建正样本，即将中文翻译到某中间语言再翻译回中文。

（1）中间语言选取：利用百度翻译api进行人工中间语言选取。由于翻译软件的中英对译过于成熟，得到的正样本的匹配数据与原数据重复度过高，可能会导致最终模型训练的容错性很低。因此我们需要找到一种二次翻译后的句子和原句子表达意思相同，并且语料重复度不是很高的中间语言。在Sentence-Bert模型支持的15种语言中，我最终选择了阿拉伯语作为中间语言。（土耳其语对语义的改动太大）

|  |  |  |
| --- | --- | --- |
| **中间语言** | **原句** | **翻译后** |
| 土耳其语 | 全会认真学习、深刻领会习近平总书记重要讲话。 | 我坐在一起，习近平秘书长真的试图谈论并理解这件重要的事情 |
| 阿拉伯语 | 全体会议认真研究，深刻理解习近平秘书长的重要讲话。 |
| 英语 | 全会认真学习和深刻领会了习近平总书记的重要讲话。 |

1. 正样本建立：利用TranslateTool.py 脚本进行文本二次翻译，生成正样本，保存在govern\_pos\_neg文件夹下的pos.tsv中。
2. 负样本建立：基于原模型，挑选其他句子中相似度最高的句子，我们的做法是将正样本数据集中的匹配方式随机调换得到负样本。
3. 将正负样本融合到一个文件train.tsv中，并标定1和0。
4. 模型测试：
5. 我们在大量模型中选取了目前性能较为优越的sBert模型。
6. 数据库建立和模型优化：

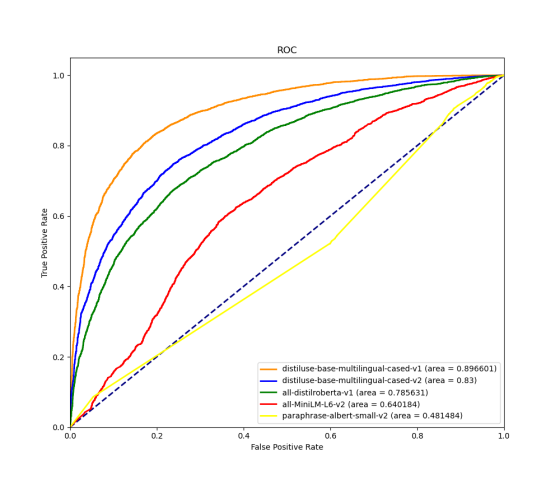
由于我们最开始选择的模型multi-qa-MiniLM-L6-dot-v1是问答模型，对于查重的语义相似度的适用性不好，对于lcqmc数据集的正样本表现较好，但是对负样本的区分度不高，因此我们并行研究模型重选取和微调对原模型进行调优。

1. 模型重选取：（我负责的部分）

在Sentence-Bert官网上，我找到十个左右的已经预训练好的模型，其中有五个以我的网络条件下可以下载并运行：

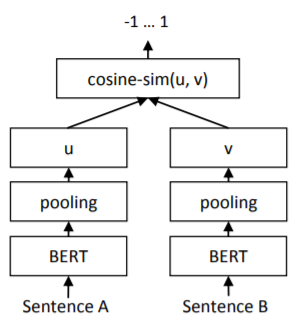
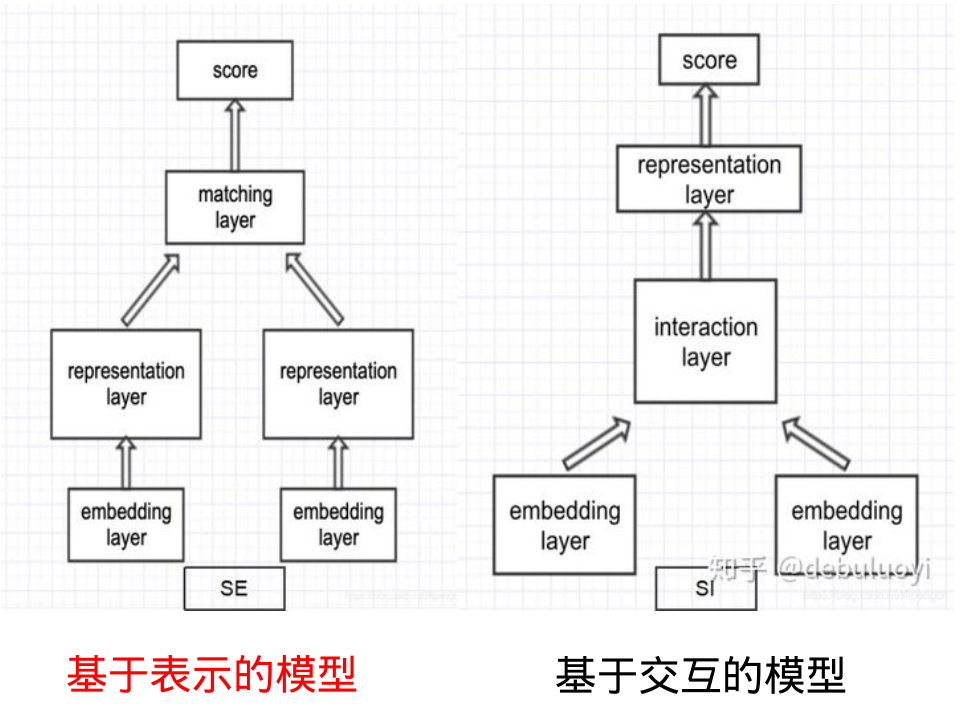
|  |  |  |  |
| --- | --- | --- | --- |
| **模型** | **大小** | **速度** | **维度** |
| Paraphrase-albert-small-v2 | 43M | 5000 | 768 |
| All-MiniLM-L6-v2 | 80M | 14200 | 384 |
| All-distilroberta-v1 | 292M | 4000 | 768 |
| distiluse-base-multilingual-cased-v2 | 482M | 4000 | 512 |
| distiluse-base-multilingual-cased-v1 | 482M | 4000 | 512 |

将这五个模型分别用于我们的数据集，发现在lcqmc数据集上的测试结果（预测准确率）如下：



因此我们选用distiluse-base-multilingual-cased-v1模型并进行接下来的微调。

这里介绍一下distiluse-base-multilingual-cased-v1模型：它是基于multilingual Universal Sentence Encoder的多语言语义相似度的模型，支持15种语言，比另一个变体distiluse-base-multilingual-cased-v2（支持50种语言）更加性能更好。



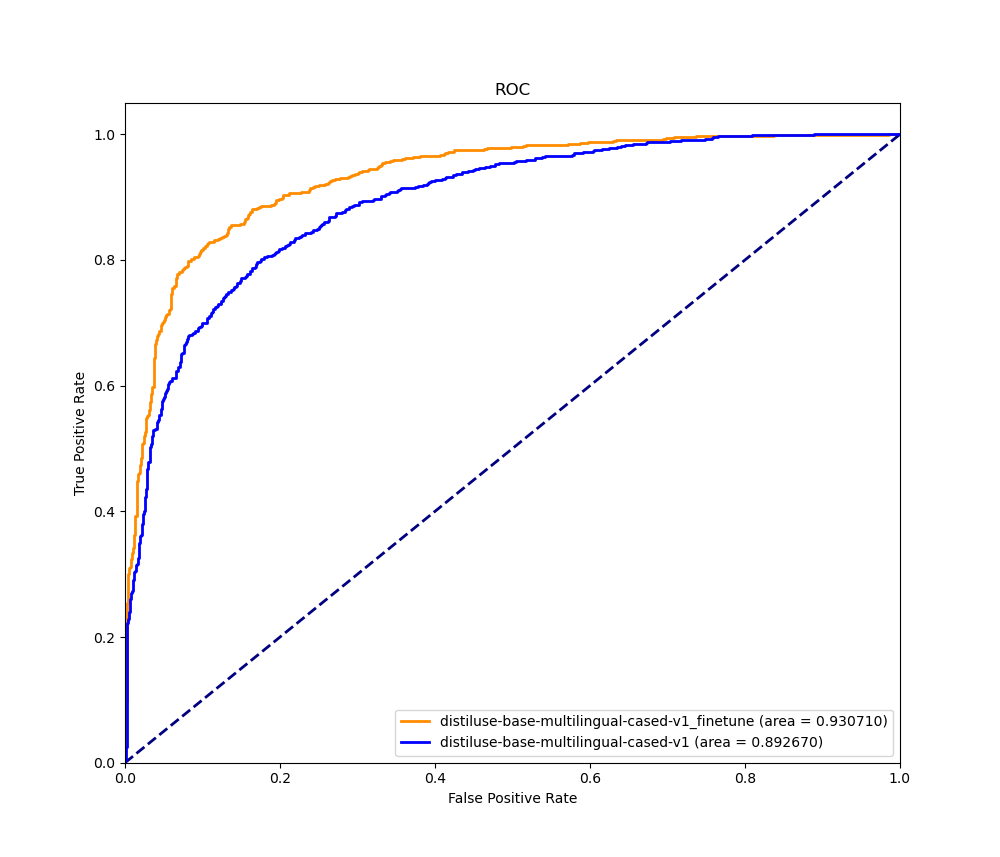
该模型是基于交互的模型，使两句词汇交互，基于交互结果计算相似度，其优势是可以捕获句子间关系，精度高。Bert是一种双向编码表示的Transformer模型，而我们使用的sBert即是以句子为单位的Bert模型，针对语义相似度这一任务的表现有了很大程度上的提升，它借助Siamese Network框架，将不同的句子输入到两个参数共享的Bert模型中，也即是同一个Bert模型中，因此将计算复杂度降低到了的数量级，极大程度上解决了原有Bert模型（将两个待计算的句子同时进入模型，进行信息交互，计算复杂度到达了）高计算量的缺点。

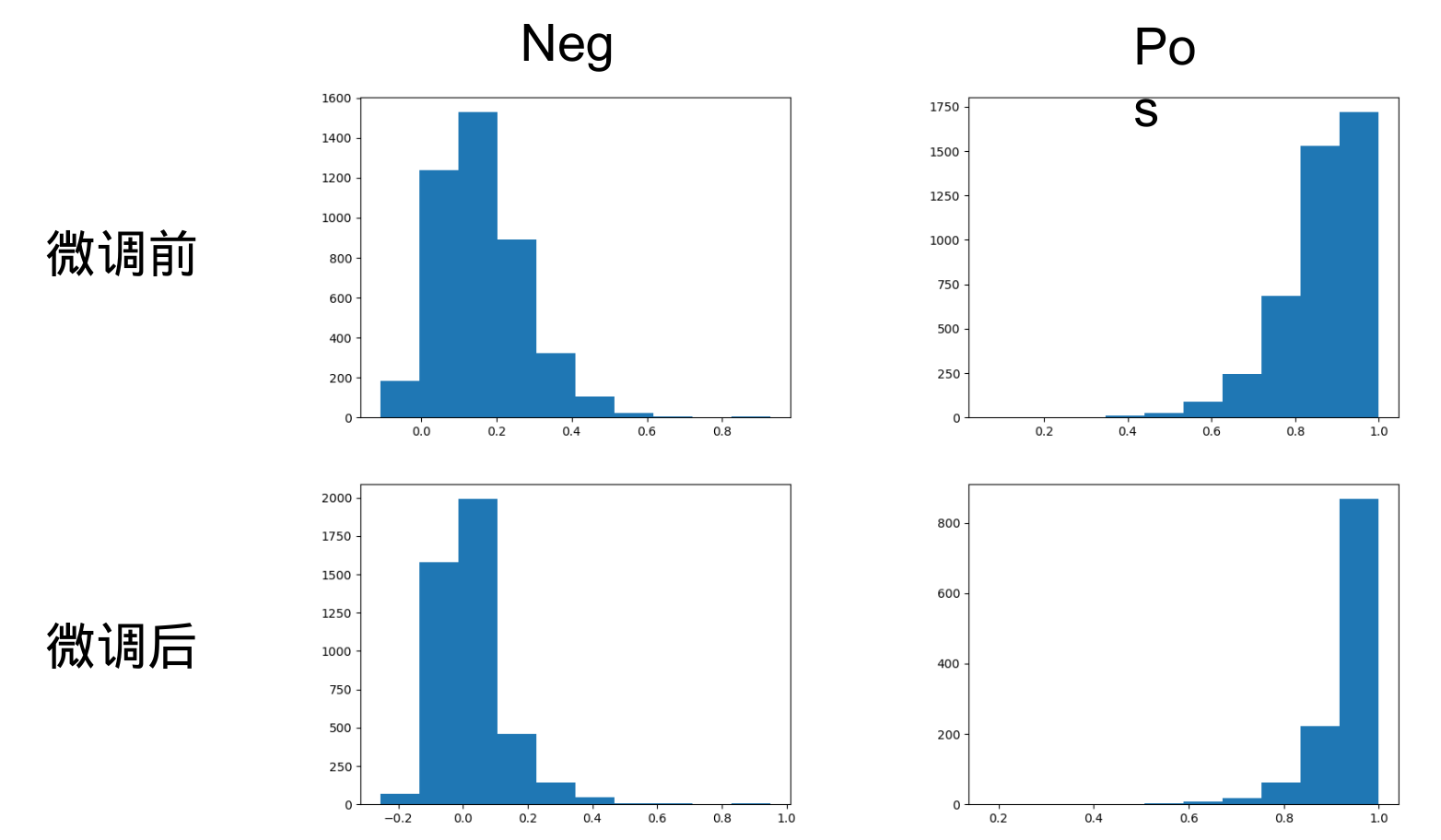
1. fine\_tune：

找到模型的开源脚本，替换掉网络最后一层的余弦相似度计 算，换用我们自己写的相似度，基于distiluse-base- multilingual-cased-v1重训一个更加适合我们政府数据集的模 型govenModel1。（微调部分我仅负责在组员微调后，将模型 进行重训）

1. 模型性能评估（我负责的部分）：

将上面预训练的distiluse-base-multilingual-cased-v1模型和我们自己微调后的模型govenModel1在政府文件数据集上进行训练，并用lcqmc的测试集进行测试（为统一不变量，因为上面五个模型的ROC性能评估是用lcqmc训练集训练，lcqmc测试集进行测试），得到distiluse-base-multilingual-cased-v1微调前后的ROC曲线图和正负样本的测试结果直方图：

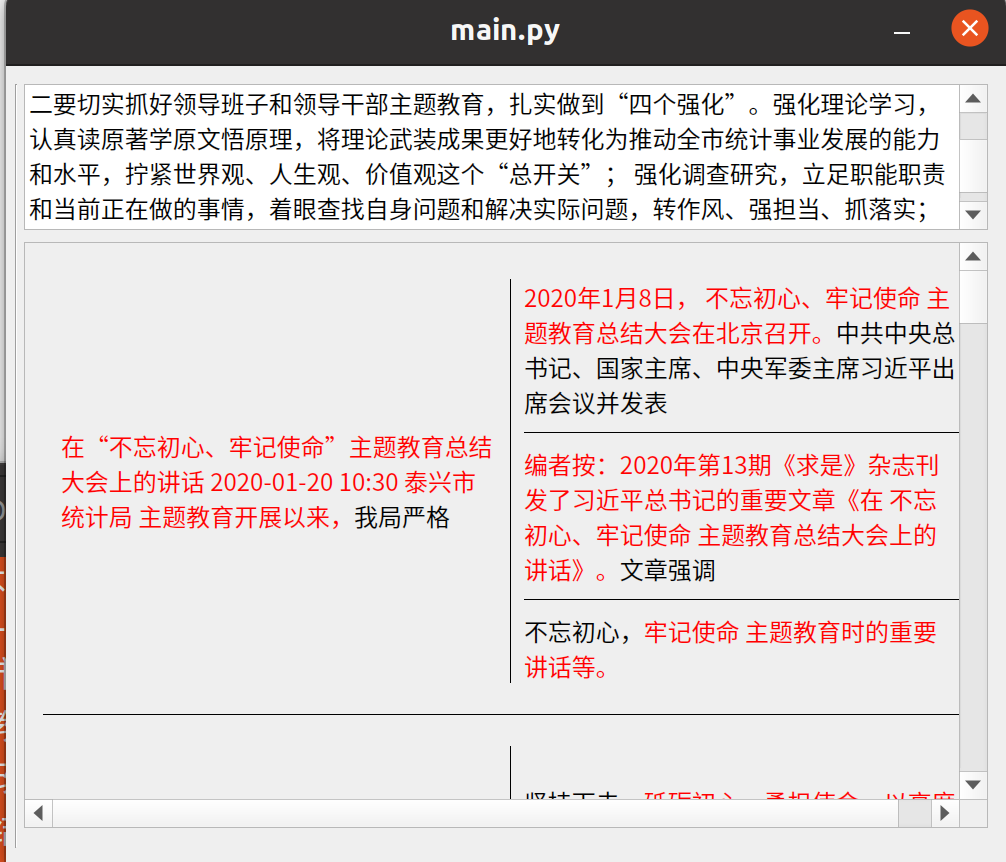


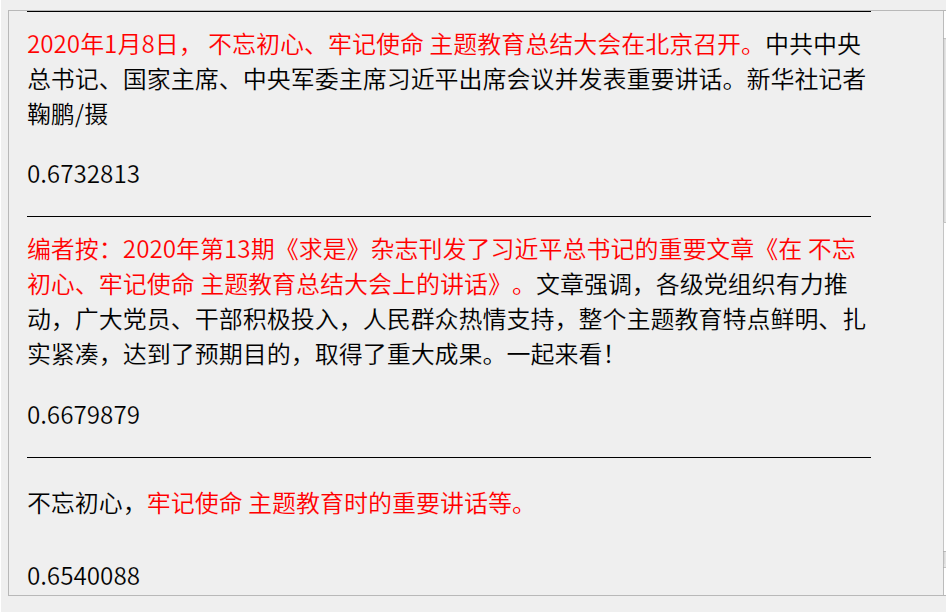


由此可见，我们的微调在政府数据集上发挥了一定的作用，即更加抑制住了负样本的测试结果，同时提升了正样本的整体预测结果。

1. 前端设计：由朱晨曦负责整体前端设计，我仅是课上在我的电脑上展示前端如何应用。
2. 实验代码：见附录。
3. **实验结果**

用government/test/1.txt进行测试：







实验结果分析：在运行main.py后，将出现一个白色对话框，将用户要进行查重的政府文件复制上去，点击左侧边栏，即可出现上面第三张图的界面，可以加入用户想要使用的中央服务器地址，点击开始查重，很快会出现第一张图的界面，标红的句子是和数据集有重复的句子，点击任何一个框，即会出现第二张图的界面，上面显示了每一段的标红句子和我们数据集句子的重复率。由于我们仅在政府文件数据集上进行模型训练，我们的软件目前只适用于政府文件的查重，之后可以将数据集扩展到更大领域，使得我们的模型可以进行更大范围上的文章查重。

1. **遇到问题及解决思路**

问题1：如何在爬虫后得到正负样本？

解决思路：以自监督方式，用一个中间语言进行二次翻译得到正样本，将正样本随机匹配得到负样本。

问题2：当模型测试准确度较低时应该怎么办？

解决思路：换用同一种网络结构的其他的预训练好的模型，并在底层模型上进行微调，来重训一个适用于自己数据集的模型。