我做的这个任务选用的数据集是一个带情绪标签的多轮对话文本数据集，训练集包括完整的情绪标签，测试集包括除了对话中的最后一句话的其他所有对话的情绪标签，情绪标签由 1 到 6 六个数字代表 Happiness、Love、Sorrow、Fear、Disgust、None 六种情绪，数据集的具体模式如图片″./大报告用图/1.png″所示。

我选择的方法是基于 Transformers，利用 BertForSequenceClassification 这个包完成分类。BertForSequenceClassification 这个包是在 BERT 模型上添加了一个线性层和激活函数用于分类。具体的实现代码是在前人的代码上进行的修改，下面我总结一下自己对代码的理解和修改的过程。

首先是导入包和设置超参数，设置一次训练中用到的数据条数（是对一次处理一条数据和一次处理全部数据的折中），模型的学习率（与模型更新的快慢有关），预训练模型的路径（预训练模型用于模型初始化和文本的序列化），处理语句的长度（模型处理的序列需要是等长的），如图片″./大报告用图/2.png″所示。

然后对随机数的生成进行限制，以便于复现，如图片″./大报告用图/3.png″所示.

然后把文本和标签同时加载成为 BERT 模型的输入数据格式，同时对文本部分进行截取和补长，使得处理的序列等长，如图片″./大报告用图/4.1.png″和图片″./大报告用图/4.2.png″所示.

然后定义正确率和 f1 值的计算函数，如图片″./大报告用图/5.png″所示。

进入到模型训练部分，加载数据，设置训练模型的设备，定义损失函数为交叉熵损失函数，定义优化器、调度器和最优模型（开始时最优模型为直接用 BertForSequenceClassification 初始化的模型），如图片″./大报告用图/6.png″所示.

然后对加载的数据逐个 batch 进行计算，得到输出，把损失值累加，根据算得的标签和实际的标签计算正确率，得到混淆矩阵，如图片″./大报告用图/7.png″所示.

然后在验证集上进行相同的处理，计算平均训练损失、平均验证损失、验证集上的正确率，根据验证集上的正确率更新最优模型，根据混淆矩阵计算准确率、召回率和假阳性值并输出，如图片″./大报告用图/8.png″所示。

最后对在测试集上进行相同的计算，并输出评价指标，如图片″./大报告用图/9.png″所示.

对原有代码进行修改之前，我首先进行了数据预处理。我建立了一个 csv 数据写入对象，先在第一列写入表头名，然后读取训练数据集，其中第一列 line[0]过滤掉，第二列 line[1]按话语间的分隔符“_euo_”切片成数组，其中每一个的元素 i-th，连同第三列 line[2]的标签集合 labels 中与自己对应的标签 labels[i-1]，一起写入 csv 文件的一行。这样训练数据集就处理完了，然后再处理测试数据集，基本采用一样的方法，唯一不同的是，如果元素是 line[1]得到的数组的最后一个元素，那么不用写入 csv 文件，而是写入另一个只有一列、每行都是待标注文本的 csv 文件。

数据预处理的过程遇到了问题，会报“string index out of range”错，根据它提示的行数，我开始时不能确定是 text_a[i]（line[1]得到的数组）还是 labels[i]（一个对话的标签集合）错了，因为我忽略了 text_a 不是 string。但我依次删除 csv_w1.writerow([text_a[i],labels[i]]) 中的 text_a[i]和 labels[i]，发现是错在 label[i]。这说明标签数不足。到现在我也没确定是为什么，可能是我的数据处理方式导致最后一行有空白行，然后这个空白行影响了，也可能是极少量数据确实存在标注缺失。因此我在处理前加入了判断，如果 text_a 和 labels 的长度不相等，就无法

正常处理，我把最近一次正常处理的数据搬到这一次来处理，等于有一个数据被处理了两次，由于异常数据可以确定很小，所以不会有影响。

然后我设置数据读取路径，运行预处理和模型两部分的代码。结果报“IndexError Target 2 out of range”的错误。我在网上查阅，发现这个错误容易由实际分类数量和代码中实际分类数量不一致导致。但是我无法确定确实如此，因为我看到的博客中的代码对模型的每一层都有清晰地定义，而我的代码中没有对模型的用到数字的设置。我首先对 read_data()函数仔细分析，但是读取数据、截断、补长都没有问题。我发现训练数据集的第一行标签正好是 2，如果真的是分类数没有设置正确，那么我把它改成 3，报错就会变成“Target 3 out of range”。我首先用代码修改，但是发现同样的代码可以生成需要样式的文件，可是替换掉这个待读取的训练数据集就不行。我又尝试在 jupyter 打开它手动修改，虽然改成功了，但是再运行模型部分代码时会报错，报错内容类似于“unexpected literal base 10”。进一步在网上查阅资料，我知道了最初的猜想是正确的，config = BertConfig.from_pretrained(bert_path, num_labels=7)来设置分类数，可是这句话在原代码中没有，我猜想可能在不设置的情况下默认是二分类。

这个模型在 4G 的 i5 8 代 CPU 上训练一趟需要接近两个小时，我设置的是趟数 epoch=5。没有 warm-up 阶段。得到的测试结果如图片″./大报告用图/10.png″所示