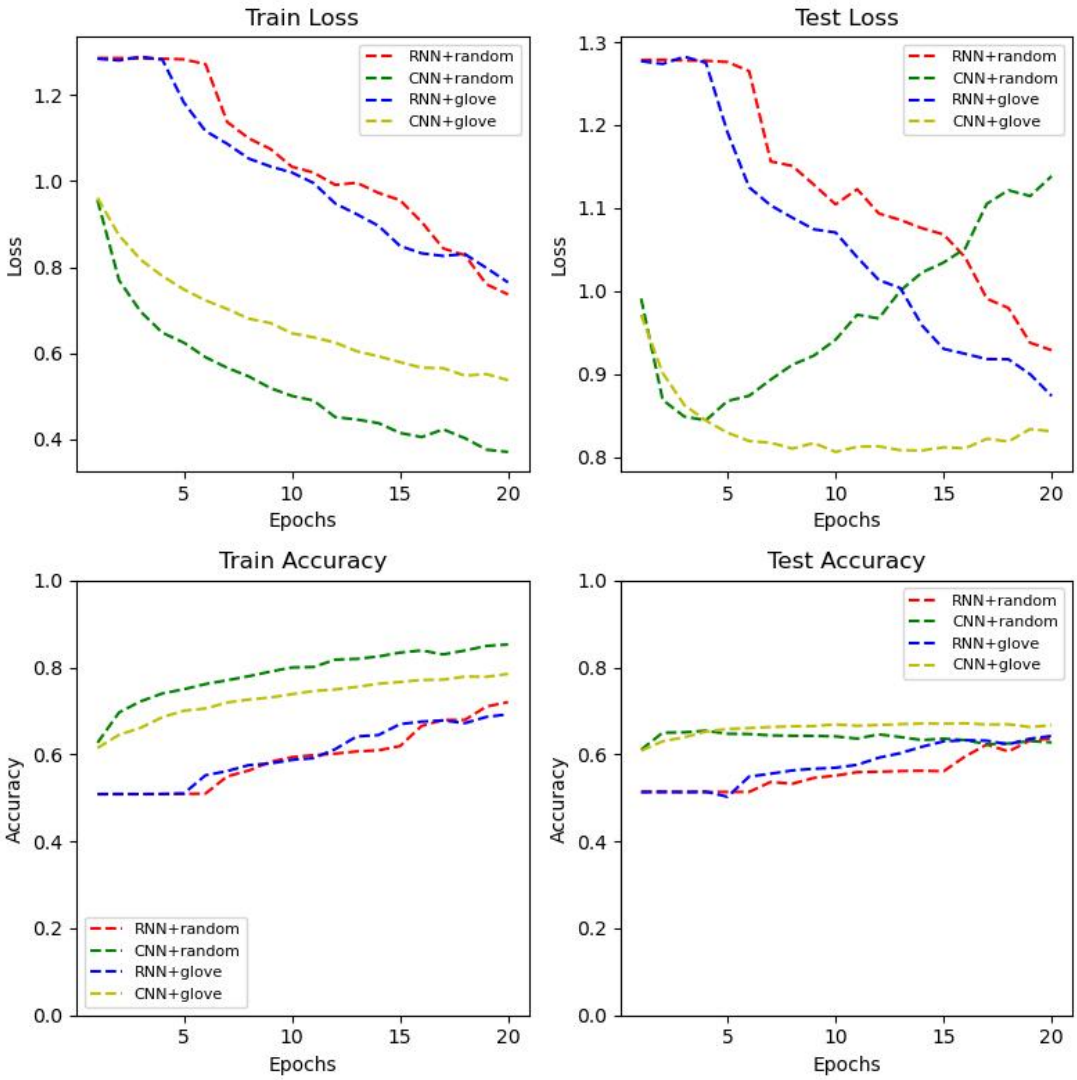
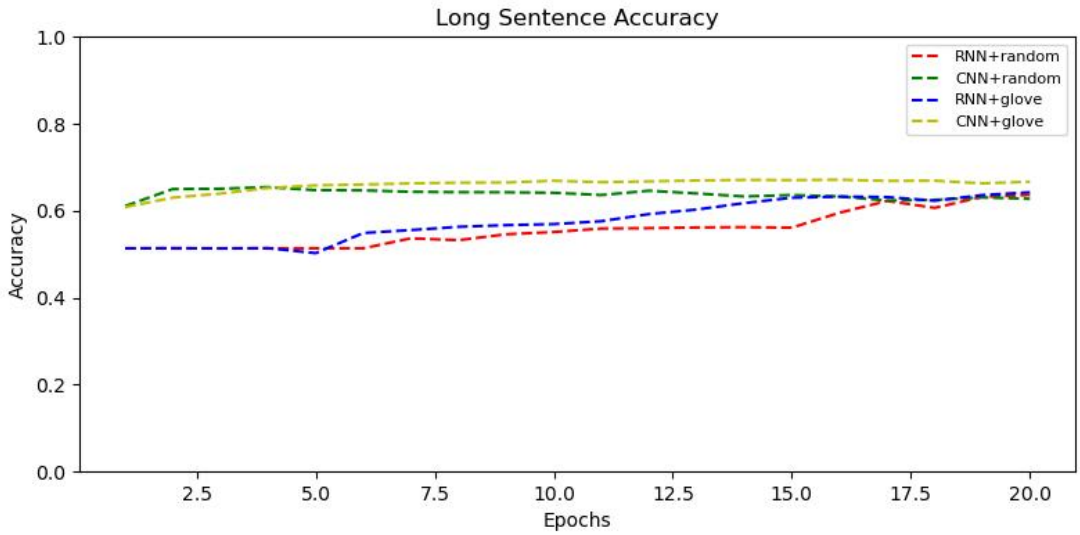
任务二：基于深度学习的文本分类

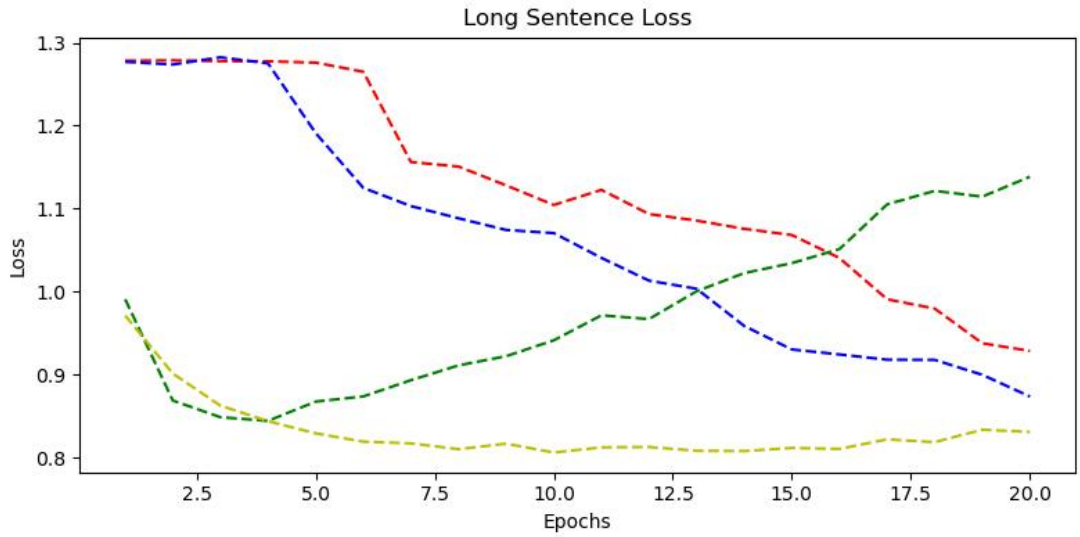
数据集：Classify the sentiment of sentences from the Rotten Tomatoes dataset
数据集的划分：kaggle 上该数据集中的 train.tsv ：70%作为训练集 30%作为测试集
2. 文本特征表示：
①Random_embedding:使用随机词嵌入
②Glove_embedding:使用 glove 预训练好的字典进行词嵌入
3. 学习方法：
CNN RNN
损失函数 直接调用 cross_entropy 交叉熵
数学方法求最优：梯度下降
4. 实验设置：
样本个数：batch_size=500
学习率：0.001
训练集：测试集=7:3
整个实验放在本地计算机的 GPU 上跑，速度还可以
5. 实验结果：



CNN+random 在训练集上 loss 几乎降为 0，但在测试集上 loss 到后面飙升，可能出现了过拟合；

总体来看，RNN 的效果没有 CNN 好，CNN+glove 的效果是最好的





在长句子上，CNN+glove 的效果也是最好的，且没有过拟合

RNN 总体效果不如 CNN，可能是因为：

RNN 是一条条地、一时间步接一时间步地处理序列，侧重于捕捉长程依赖。但在很多情感、话题分类里，关键信息其实集中在几个短语或关键词，RNN 在这方面反而“过度”——它要维护整个序列的状态，往往更容易出现梯度衰减或信息稀释。

而 CNN 中的池化可以捕捉到句子中比较关键的词或句子，从而一针见血地完成分类。