模式识别与机器学习实验三——循环神经网络实现

2021113117-王宇轩

1. 实验环境

- 操作系统: Windows
- 编程语言：Python
- IDE：PyCharm
- 设备：GPU

2. 文件列表

文件名	内容
mymodels.py	自定义模型、文本数据集、文本分类的训练和测试函数库
text.py	进行文本分类的源程序
climate.py	温度预测源程序（调用了mymodels中的MyLSTM）
log.zip	五个最好模型的训练loss日志文件
实验报告.pdf	实验报告

3. 实验内容

3.1 文本分类数据集的处理

使用了腾讯ailab的预训练词向量，维度为100：
https://ai.tencent.com/ailab/nlp/data/tencent-ailab-embedding-zh-d100-v0.2.0-s.tar.gz。
以及哈工大停用词表：
https://github.com/goto456/stopwords/blob/master/hit_stopwords.txt。

大致流程为，读入csv文件后，删除label列，将cat列文本映射为类别编号数字。对于review文本，使用jieba分词并根据停用词表排除停用词。按要求划分训练集、验证集和测试集，并**还原索引**。用gensim库的KeyedVectos模块加载词向量并加载为embedding层。这一部分的具体代码在**text.py**中。

在**model.py**中定义了自定义数据集类`MyDataset`，其中作了划分自变量和因变量、将自变量（词列表）根据`word2id`映射为词id并进一步通过embedding层替换为词向量、导出为张量的工作。

3.2 气候数据集的处理

在**climate.py**中完成。主要流程为：读入csv文件，保留日期时间和温度列，用字符串匹配完成根据年份划分训练集和测试集。自定义数据集类`CliDataset`，在其内部按照要求根据天数划分自变量和因变量，并转换为张量。

3.3 模型实现

见**mymodels.py**，大致思路均为继承`nn.Module`类，并用`nn.Parameter`来定义模型内部权重矩阵，并根据网络结构编写`forward`函数。
此处，我一开始一直遇到训练时损失函数NaN值（加上梯度裁剪和用Adam优化器甚至更为严重）、训练后的模型精度很低的问题，原来是用nn.randn()来随机初始化权重参数的时候，要*0.01调整方差。调整后损失函数异常值现象不再出现，精度也能正常上升。

3.4 模型训练、测试

对于文本分类，模型训练、测试函数在**mymodels.py**内。通过修改**text.py**中模型`model`定义时选用的mymodels（mm）中的模型来改变选用的模型。

对于温度预测，都在**climate.py**中实现，只是引用了**mymodels.py**中的`MyLSTM`模型。

3.5 结果输出

见源码。对于温度预测，**climate.py**会随机选取四组测试集中的数据，绘制7天的真实和预测曲线图。

4. 实验结果及分析

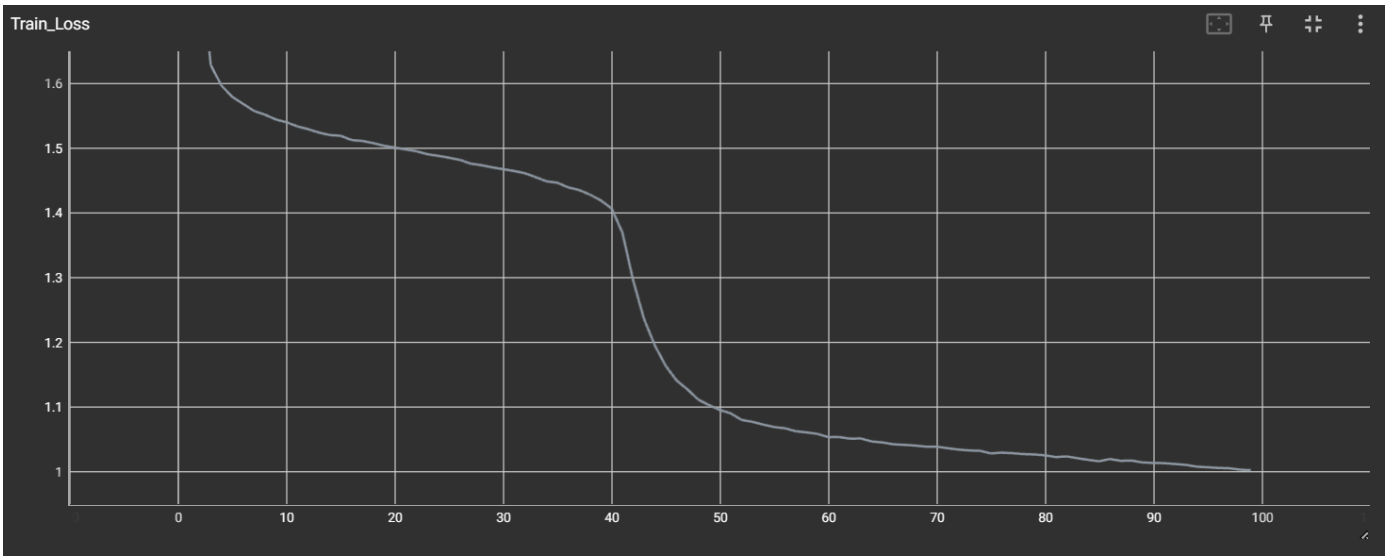
以下依次给出了各次获得了最佳结果的run的超参数、损失函数曲线和结果。和实验二一样，也经过了漫长的调试调参，而且由于训练时间更久、模型更复杂，比实验二痛苦许多，也因此即使其实可以看到许多模型如果继续训练还有性能提升的空间，获得了还可以的结果就没有继续训练了。

4.1 文本分类

RNN

```
sentence_max_size = 50
batch_size = 64
num_epochs = 100
lr = 0.1
hidden_size = 512
```

选用优化器为SGD。

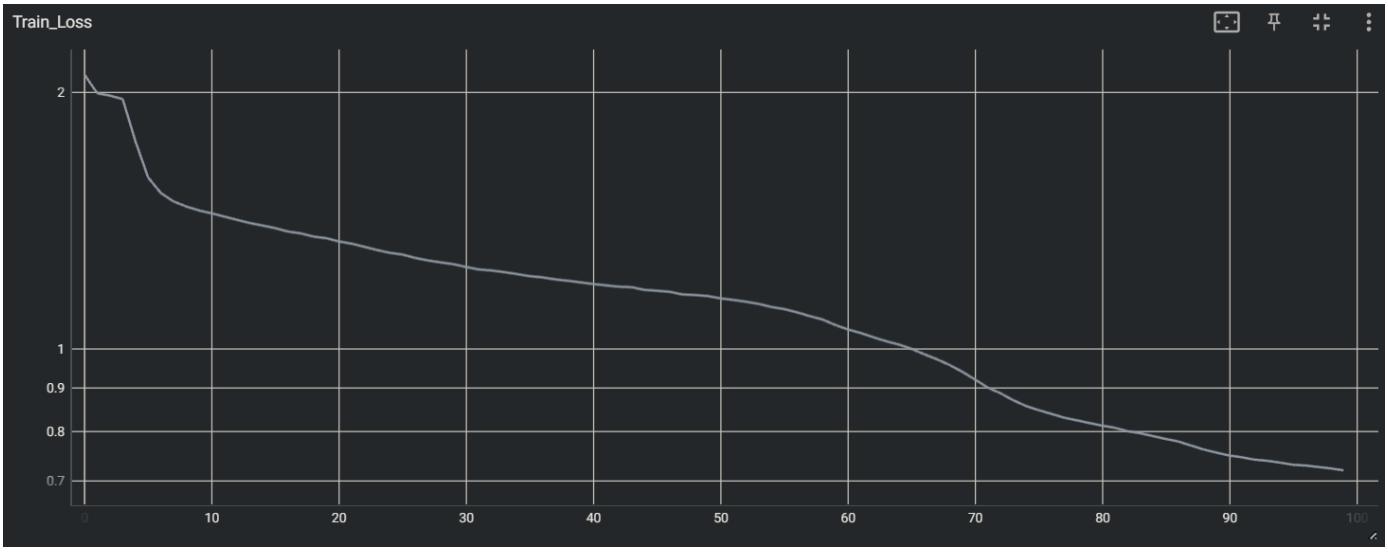



```
训练集测试精度：0.6065，召回率：0.4381，F1：0.4316
验证集测试精度：0.6109，召回率：0.4421，F1：0.4365
测试集测试精度：0.5996，召回率：0.4350，F1：0.4293
```

GRU

```
sentence_max_size = 50
batch_size = 64
num_epochs = 100
lr = 0.1
hidden_size = 512
```

选用优化器为SGD。

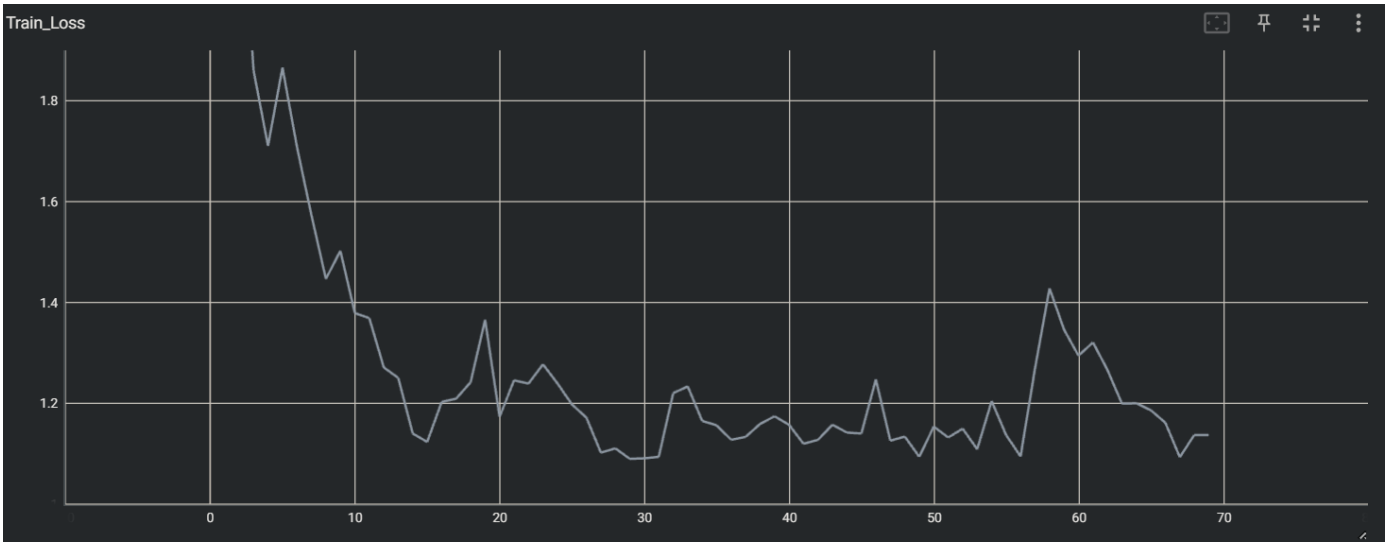



```
训练集测试精度：0.7634，召回率：0.6533，F1：0.6628
验证集测试精度：0.7668，召回率：0.6559，F1：0.6645
测试集测试精度：0.7605，召回率：0.6522，F1：0.6634
```

LSTM

```
sentence_max_size = 50
batch_size = 64
num_epochs = 70
lr = 0.1
hidden_size = 512
```

选用优化器为Adam。

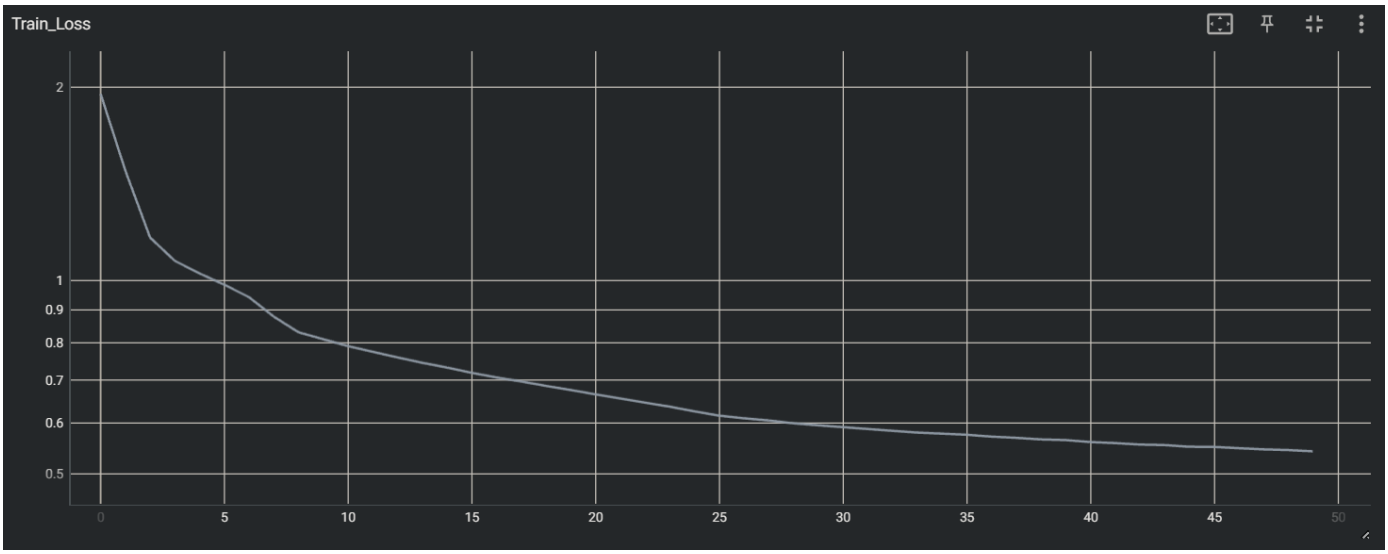



```
训练集测试精度：0.7307，召回率：0.6569，F1：0.6337
验证集测试精度：0.7246，召回率：0.6539，F1：0.6288
测试集测试精度：0.7260，召回率：0.6524，F1：0.6286
```

Bi-RNN

```
sentence_max_size = 50
batch_size = 64
num_epochs = 50
lr = 0.1
hidden_size = 512
```

选用优化器为SGD。

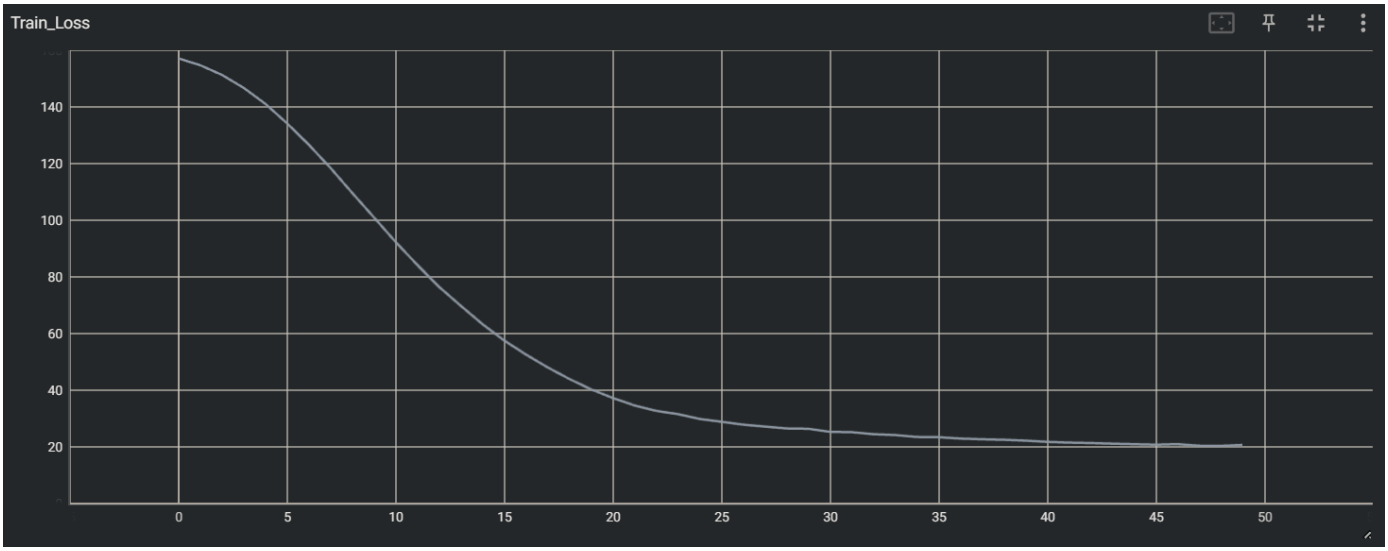



```
训练集测试精度：0.8108，召回率：0.7079，F1：0.7192
验证集测试精度：0.8049，召回率：0.7016，F1：0.7130
测试集测试精度：0.8069，召回率：0.7050，F1：0.7164
```

4.2 温度预测-LSTM

```
batch_size = 64
num_epochs = 50
lr = 1
hidden_size = 512
```

选用优化器为SGD。




```
Batch1：平均误差2.9434，中位误差2.1798
Batch2：平均误差3.1754，中位误差2.4485
```

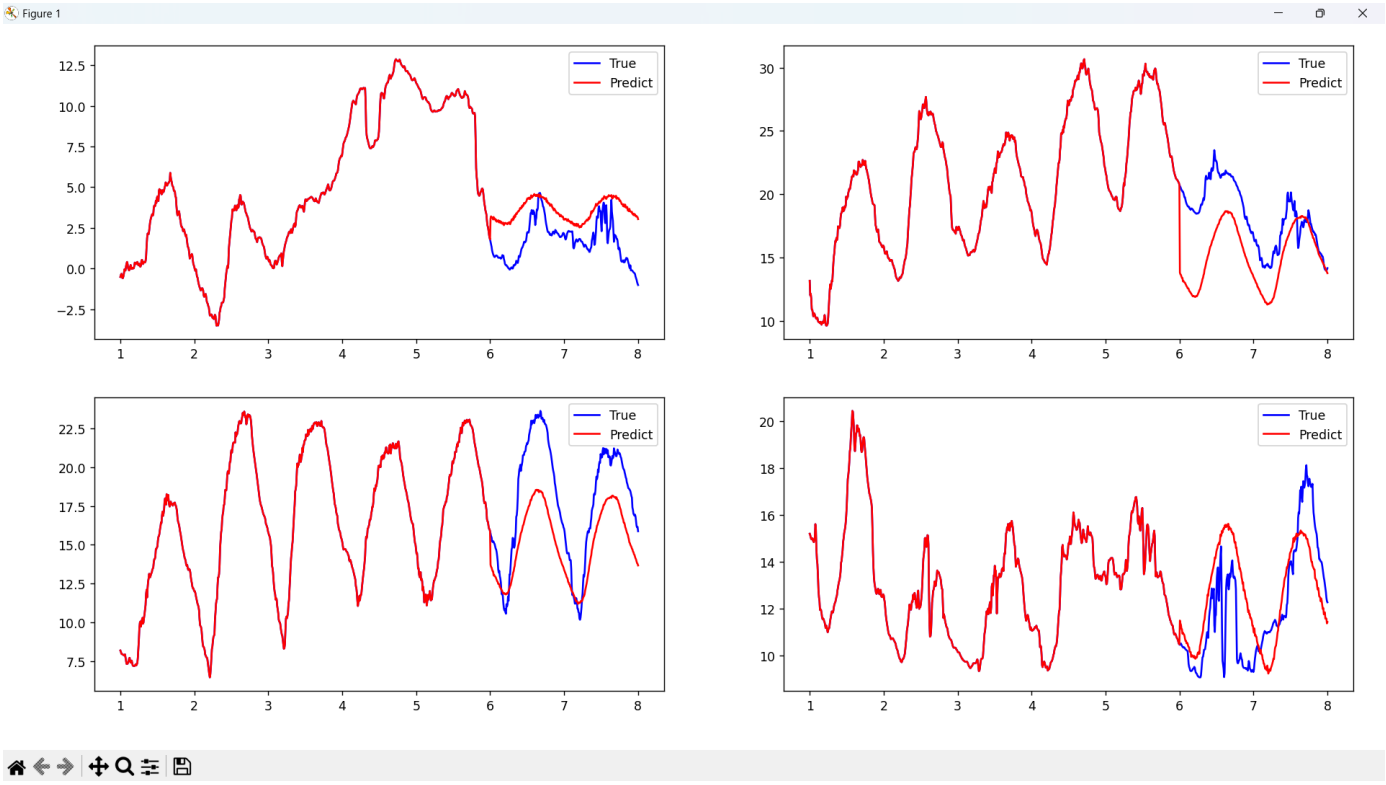


Figure 1
True
Predict
True
Predict
True
Predict
True
Predict