

# 2022-11-01-Domain\_Adaptation总结

---

## Domain Adaptation复现(总结)

本实验基于传统的Domain Adaptation架构进行复现。

### 数据处理：

- train vs test:
  - subject independent: 32个人十折交叉验证
  - subject dependent : 对每一个受试者进行十折交叉验证，最后32\*10取平均作为最终结果
- data preprocessing:
  - 1s window: 对于每一个trail，将60s按照1s no-overlap window切分，切成60个样本
- Base Feature Extraction:
  - DE: 首先将数据滤波到4个频带，然后分别对4个频带取DE特征。
- Summary:
  - 数据的构成为: (None, 32, 4) # （样本数， 脑电通道， 特征数）
- 标签： 暂时只在valence维度进行实验

### Models：

- Features Extractor:  
双层LSTM
- Label Predictor:  
双层Dense
- Domain Predictor:  
双层Dense
- Loss:  
BinaryCrossEntropy
- Optimizer:  
Adam

# An Efficient LSTM Network for Emotion Recognition from Multichannel EEG Signals

针对该论文的问题及相关解决措施

- Base Feature: 本文采用DE提取特征, 并且将一个trail切分成60s, 每一个窗口提取特征, 然后将60个窗口进行了拼接, 这也就意味着在一次subject dependent中, 只有39个样本作为训练集, 1个样本作为测试集。样本 shape(None, 32, 4\*60)
  - 原文: The sizes of DE features for 1-second windows are (32, 4), (62, 5) and (30, 5) for DEAP, SEED and CMEED databases<sup>1</sup>, separately. We compute DEs of all windows and concatenate them to form a feature vector for representing one sample.
  - 修改: 修改为传统的1s窗口做一个样本
- Label Predictor中,  $W_{\{a, c\}}$  参数未定义, 也就是在Attention中, Query未定义。
  - 修改: 去掉Attention模型, 转为验证Domain Adapatation模型的有效性。
- 实时更新的: 由于模型将源域和目标域分别设计为训练集和测试集, 所以如果来一个新的测试集还要重新训练
  - 不解决。。。

## 总结

## 模型

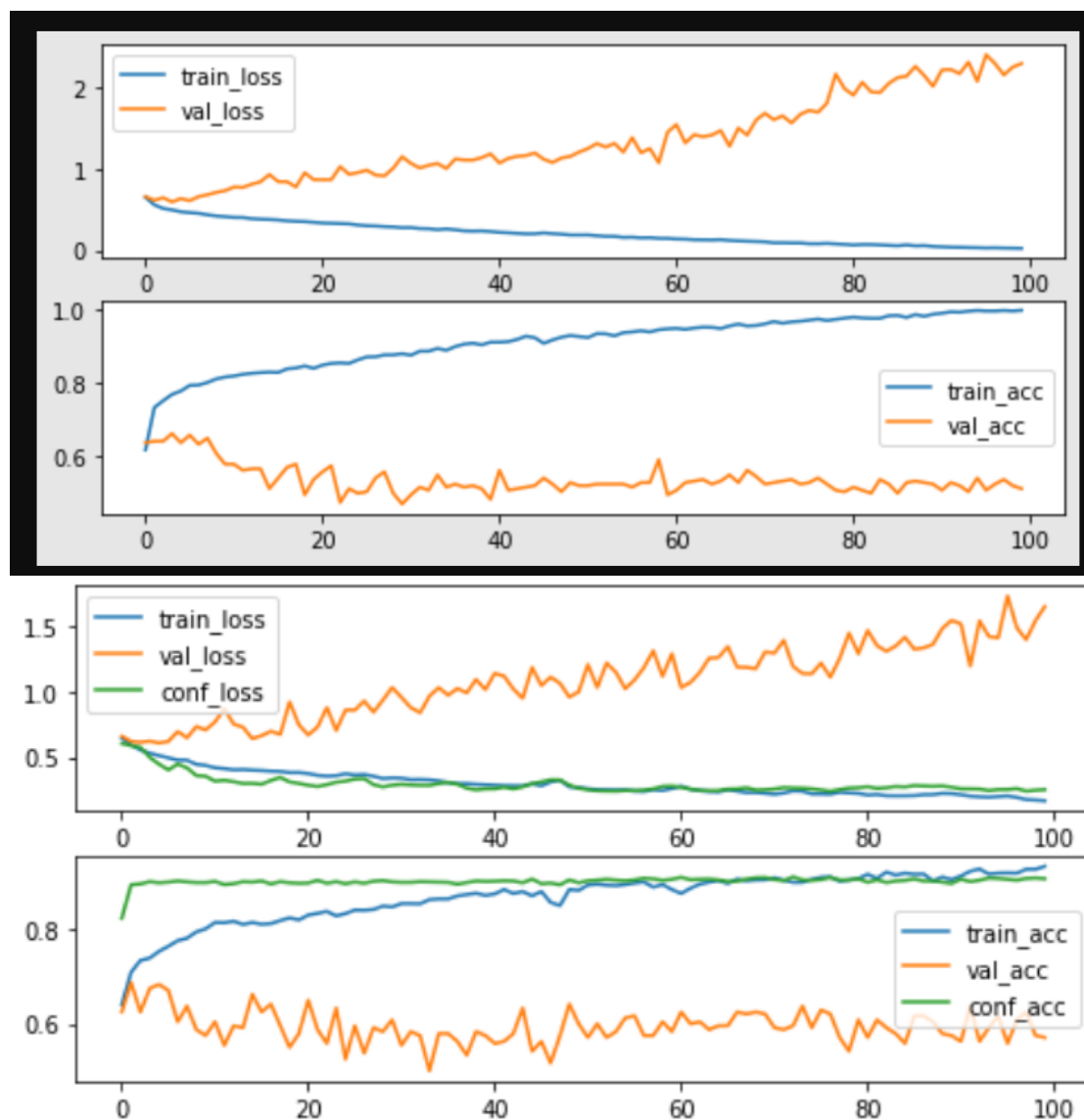
```
def FeatureExtractor_1():
    return tf.keras.models.Sequential([
        tf.keras.layers.LSTM(16, input_shape=(32, 4),
return_sequences=True),
        tf.keras.layers.LSTM(32, return_sequences=True),
        tf.keras.layers.Flatten()
    ])
def DomainPredictor_1():
    return tf.keras.models.Sequential([
        tf.keras.layers.Dense(64, activation='relu'),
        tf.keras.layers.Dense(2, activation='softmax')
    ])

def LabelPredictor_1():
    return tf.keras.models.Sequential([
        GradientReversalLayer(),
        tf.keras.layers.Dense(128, activation='relu'),
        tf.keras.layers.Dense(2, activation='softmax')
```

## 针对subject dependent

subject dependent指，每一个受试者单独一个模型训练。

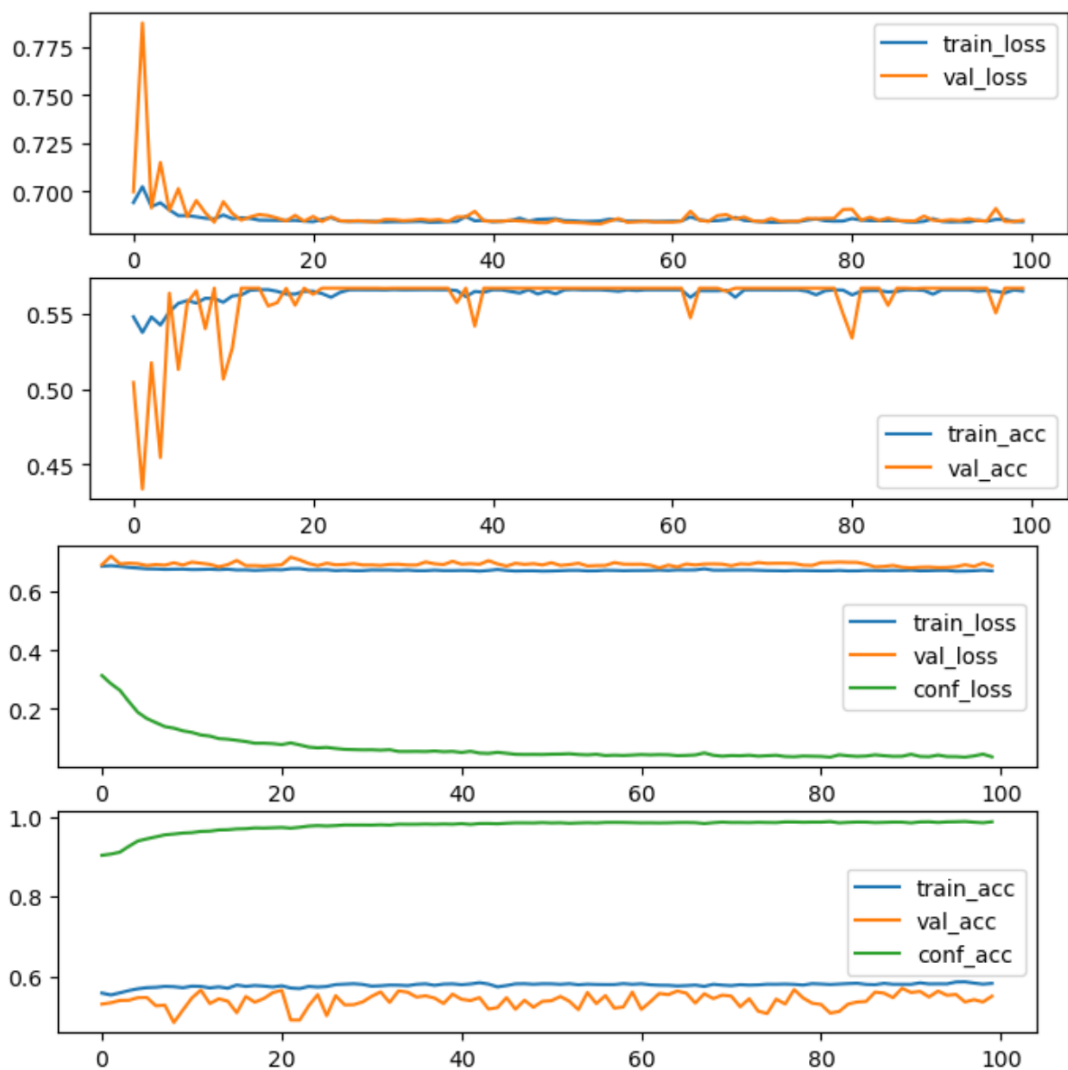
基础模型与DANN模型如下：



可以看出，使用DANN模型并没有过多的提高测试集的分，而且从理论上讲，使用DANN模型中，`conf`(训练集和测试集的混淆分数)应该是下降的，但是实际上，这个分数越来越高，这也就意味着模型并没有起到他应有的作用。

## 针对subject independent

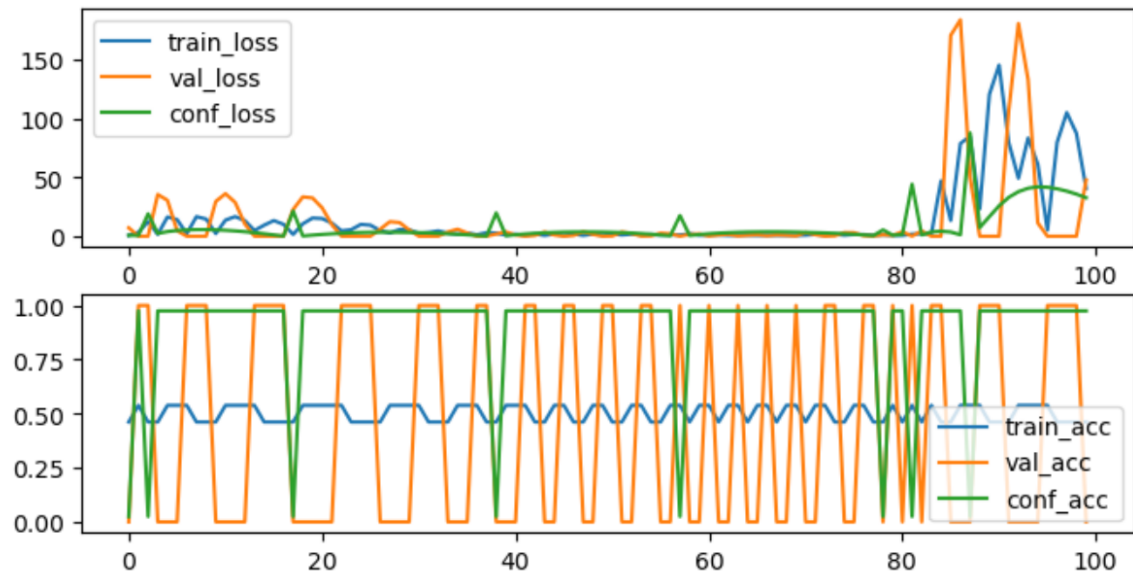
subject dependent在此处指，按照受试者分类划分



图中可见，DANN中conf的准确率和loss非常好，但是训练集和测试集的效果就没有想象中优秀。而且相比于base完全没有进步，base的结果是0.566，DANN的准确率是0.551

## 针对论文中，提出的按照39VS1进行实验。

在这个部分只做了受试者1的一次实验，实验结果如下：

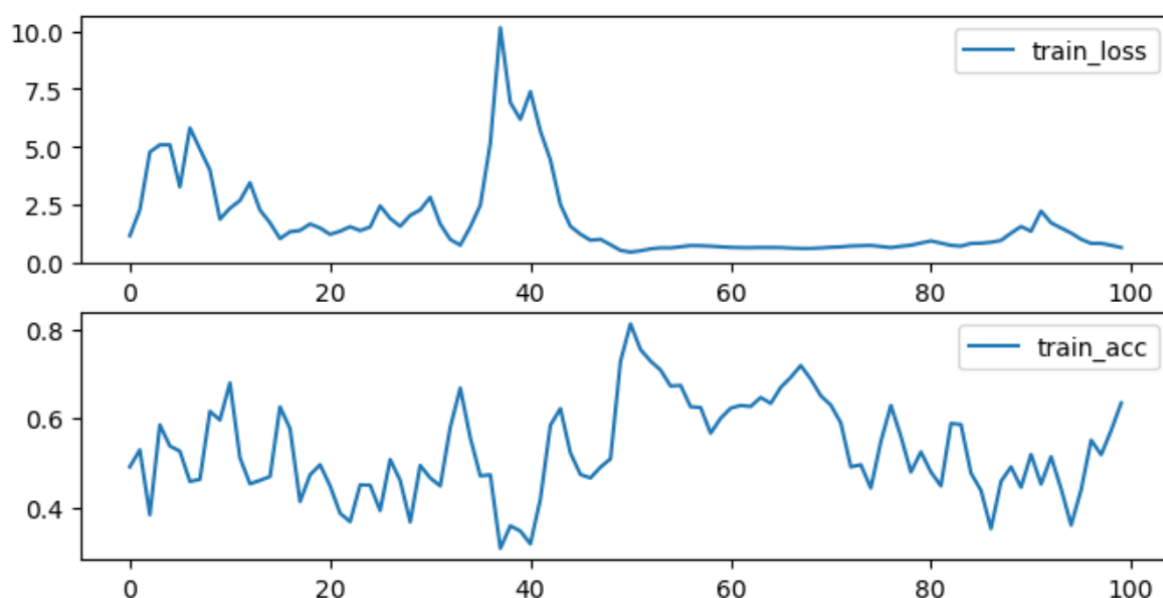


图中可以看出，有一个非常奇怪的现象，那就是在本轮实验的最后几个epochs，loss反而开始波折起来了，这个现象暂时不知道怎么解释。

另外，可以看出，val\_acc准确率一直在0/1上下波动（因为就一个样本），所以我合理怀疑是否有可能论文中的.90准确率是因为恰好波动到1的次数多>，而和模型的优劣无关。

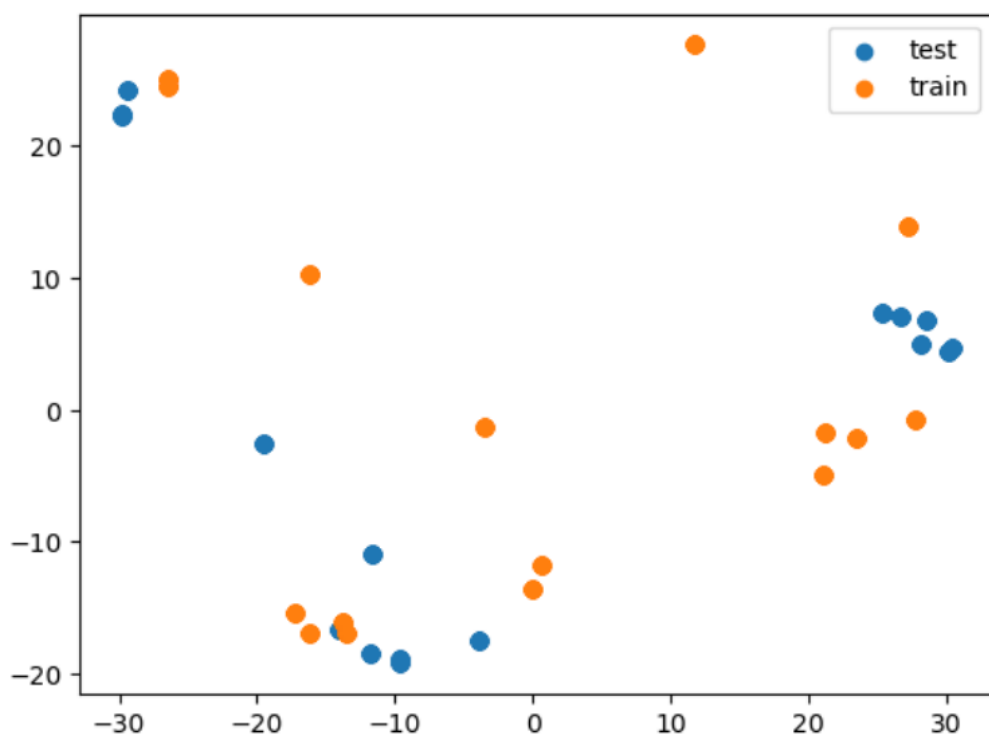
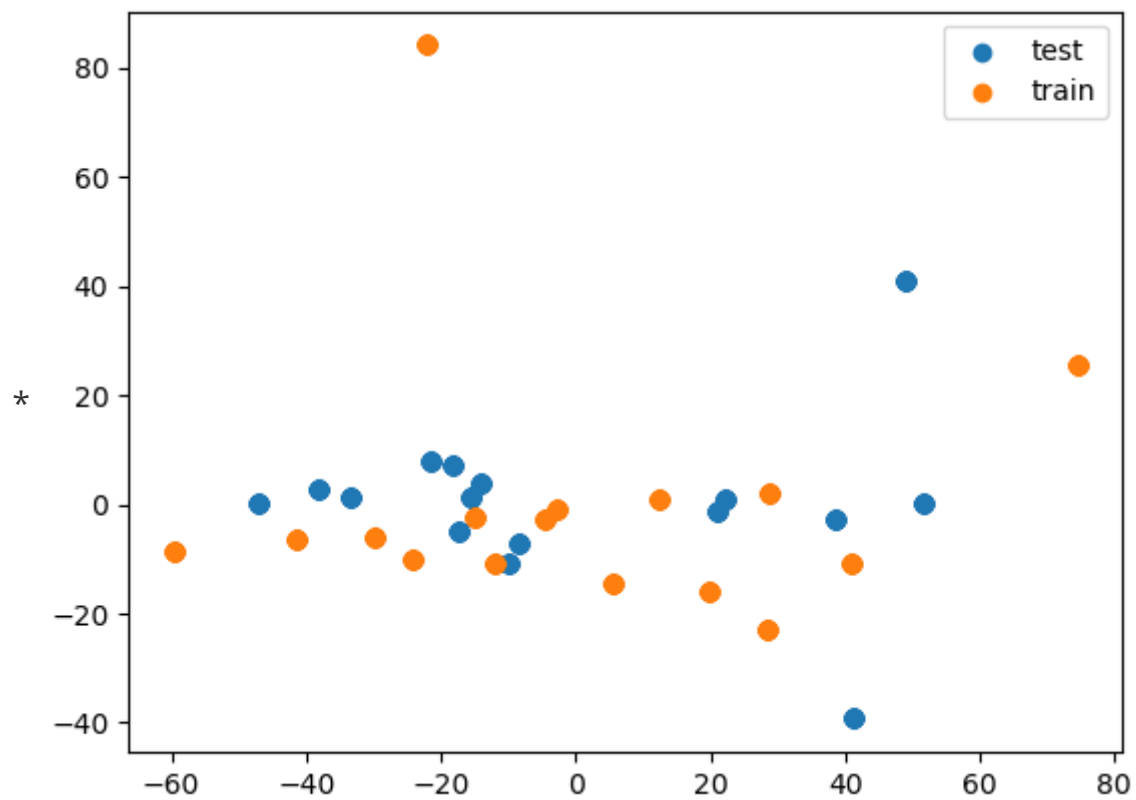
## DANN有效性分析。

有效性分析，主要是应用在subject independent上。在这个实验中，随机将受试者分为两个部分。一部分作为测试集，一部分作为训练集。此外，模型只使用feature generator和domain predictor。想要判断以下，是否真的有效。



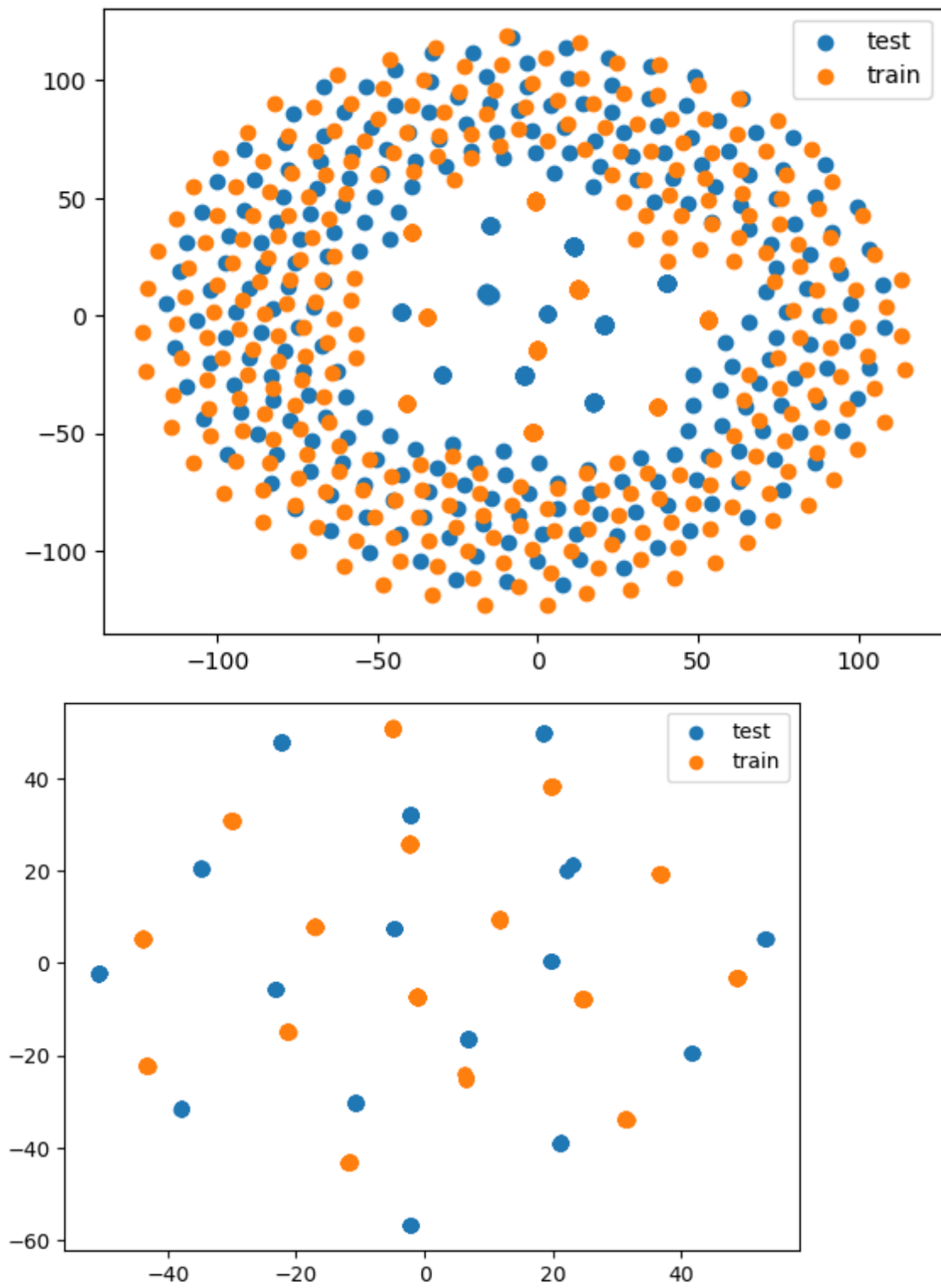
loss和acc的结果似乎么有表现出什么规律。

经过PCA降维后的结果与经过模型训练后降维的结果



大部分样本都是重合的，而且经过模型训练后，这个结果似乎并没有出色到哪里。

经过TSNE降维后的数据和经过模型训练后降维



TSNE降维后的结果就更明显了，基本上就是趋同的。但是train和test的同分布，感觉在反而原始特征中更能体现