MovieLens期末作业

——跨模态数据对齐

作业说明

- **作业分数：**本次作业选择性完成，作业分数可以代替期末考试中的一道与深度学习相关的大题的分数
- **作业背景：**期中作业的“遗留问题”之一，电影海报内容与电影简介内容的相关性探索
- **作业任务：**利用pytorch搭建神经网络，实现电影海报内容与简介内容的对齐匹配
- **作业目标：**体会基于神经网络进行数据分析的整体流程
  - 任务建模、数据预处理、网络设计、训练与测试、结果分析...

任务说明

- 对共计N张海报和N段简介文本，定义它们的“距离“矩阵$D \in \mathbb{R}^{N \times N}$
- 其中D_{ij}表示第i张海报和第j段文本之间的距离

- 我们的目标是使得对应于同一部电影的海报和文本之间距离最小，也即使得$\arg\min_j D_{ij} = i$且$\arg\min_i D_{ij} = j$

- 当然，要让每一对匹配的海报和文本距离都能取到最小非常困难，本次作业的评价指标中只需要让这个距离位于该海报/文本与所有文本/海报距离最小的前5%即可

任务说明

- 评价指标
- top-k acc

```python
def get_acc(D, ratio=0.05, dim=1):
    '''
    Calculate the accuracy based on the top-k nearest neighbors in the distance matrix.

    Parameters:
    D : torch.Tensor
        The distance matrix with shape (N, N), where
        D[i, j] represents the distance between poster[i] and intro[j].
    ratio : float
        A float in the range (0, 1) that determines the proportion of nearest neighbors to consider.
    dim : int
        The dimension along which to compute the top-k neighbors.
        Use dim=1 for top-k intros for each poster, dim=0 for the top-k posters for each intro.
    '''

    total_samples = len(D)
    k = int(ratio * total_samples)

    _, topk_indices = D.topk(k, dim=dim, largest=False)
    if dim == 0:
        topk_indices = topk_indices.T

    correct_matches = 0
    for i in range(total_samples):
        if i in topk_indices[i]:
            correct_matches += 1
    accuracy = correct_matches / total_samples
    return accuracy
```

TASK #1 数据处理与数据集搭建

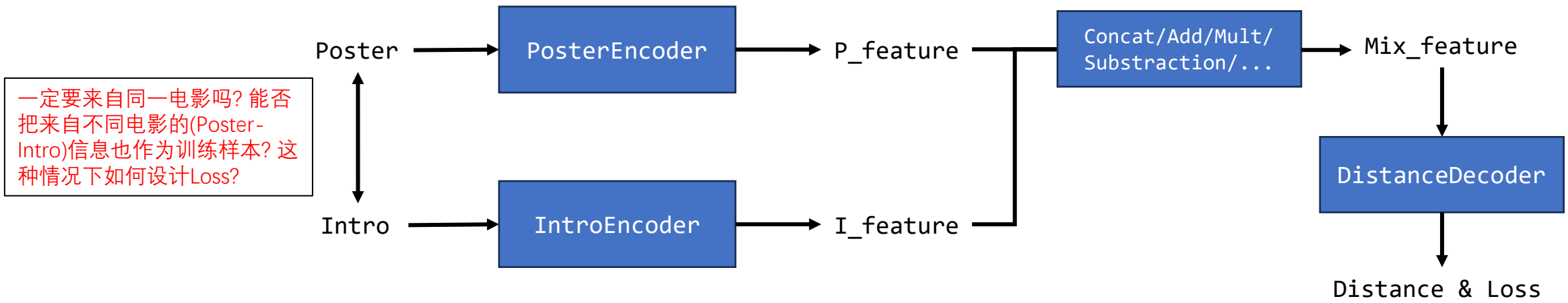
- Step 1. 数据筛选。部分电影没有对应的海报，在数据读取时应筛除这部分电影的数据，最终剩余的电影数目应为N=2938.
- Step 2. 数据预处理。使用合理的方式预处理数据（特别是简介文本），文本数据的一些预处理方法可以参考HW09
- Step 3. 数据集搭建。使用60%的数据作为训练集，**但与通常的分类任务不同的是，我们使用全部100%的数据作为测试集**
  - 为尽量避免训练集划分的随机性带来的性能差异，作业环境中提供了train.txt文件，内含作为训练数据的电影id，请不要自行划分训练集

TASK #2 模型设计与训练

- Step 1. 模型设计
  - 海报特征提取: CNN、Traditional Features + Linear Layer、...
  - 简介特征提取: TFIDF、RNN/LSTM、nn.Embedding、Pretrained Word Vector、Pretrained Language Model、...
  - **注:** 如需在简介特征提取中使用基于词频统计、TFIDF等方法，只能使用训练集中的简介数据
- Step 2. 训练流程
  - 问题建模设计、损失函数设计、距离函数设计、...
  - 训练tricks: Optimizer、LR Schedule、...

TASK #2 模型设计与训练

- 一种简单的(训练过程)建模方式



- 也许会有所启发：CLIP （原论文已随作业环境发布）
- **仅作参考，可以积极探索其他建模形式**

TASK #3 测试与分析

- Step 1. 使用训练完的模型在全数据上运行，得到形状为N*N的距离矩阵D，再使用get_acc函数，取ratio=0.5,dim分别为0,1得到两个acc，作为评测结果
- Step 2. 选取一些海报/简介，利用你所得到的距离矩阵D挑选出与之距离最接近的top-k（k可自选）简介/海报，尝试对以下问题进行分析
  - （在这一数据集上）电影海报与电影简介内容是否具有某些相关性?
  - 对于给定的海报/简介，最接近的前K个简介/海报是否共享相似的语义内容、主题、风格、或其他特征?
  - 跨模态对齐的特征可能在哪些下游任务上发挥怎么样的作用?
  - （均为开放性问题，回答合理即可）

评分方式

本次作业选择性完成，作业分数可以代替期末考试中的一道与深度学习相关的大题的分数（笔试分和作业分取**max**）

- 基础部分（60）：
  - 代码能跑通、并且代码完成的功能与作业要求一致（50）
  - 没有关键性错误和“作弊”现象（如用测试数据进行训练、在距离矩阵D的对角元上偷偷减去一个值、在测试时修改ratio的值等）（10）
- 评测部分（15）
  - get_acc中分别取dim=0和1，得到范围在[0,1]的acc1和acc2，测评部分得分为

    ```
    10 * min(acc1+acc2, 1.5)
    ```

  - 如果在代码中“作弊”，本部分不得分
  - 满分有难度，没必要硬卷，最终视大家的acc情况做调整
- 分析与探索部分（25）
  - Task 3 Step 2 的问题分析（10）
  - 在问题建模、模型设计、损失函数和距离函数、特征提取、预处理等某个或某些方面，(1)进行了多样性尝试并**对尝试结果有具体分析**，或（2)有出彩的设计（如期中作业里陈锐韬同学的问题建模方法）(15)

提交说明

- DDL： 6.1晚23:59
- 需要提交的文件包括
  - 源代码： ipynb要求能从头直接运行，并且**保留了单元格输出结果**
  - 报告： 要求**PDF**格式
  - 模型： 如果训练部分代码在GPU上运行超过10min，请将模型参数保存并提交，**并在ipynb文件的测评代码前加一个读取保存的模型的单元格**
  - 其他可能的必要文件： 如在文本处理中的停用词表、预训练词向量文件、包含有用的类/函数的py文件等
- 以上文件压缩为一个“**学号.zip**”提交
- 如果文件太大，教学网提交不方便，可以放**北大网盘**并在教学网提交共享链接

作业提交格式不符合要求的，会有少量扣分