深度学习课程作业拟定

* **小作业（40分）**

本课程的小作业旨在使同学们熟悉并入门国产的开源深度学习框架（Jittor, Paddle, Mindspore），包括但不限于安装框架，配置深度学习环境，设计实现简单的深度学习模型，按照问题需求对模型进行修改等。目前计划安排小作业两次，难度逐步增加。同学们需要在DDL前将小作业相关的文件（包括报告和代码）一起打包上传到Canvas上。（超过DDL十天以内，晚交会扣除一定的分数，每晚交一天扣一分，作业超过十天则不再收。）

**第一次小作业：乘法器（10分）**

目标：

1. 安装国产深度学习框架 Jittor，Mindspore 或 Paddle （三选一，推荐Jittor）
2. 配置工作环境
3. 训练一个简单的模型，目标是利用神经网络实现乘法操作

安装教程：

1. Jittor（计图）：<https://cg.cs.tsinghua.edu.cn/jittor/download/>

2. Mindspore（昇思） : <https://www.mindspore.cn/install>

3. Paddle（飞桨）: <https://www.paddlepaddle.org.cn/install/quick>

任务：

我们知道，神经网络中大部分是线性操作，仅有的非线性操作也可以是ReLU这种分段线性函数，那么，如果用神经网络去拟合乘法操作，效果会怎样呢？

要求：

本次作业要求自行设计一个深度学习模型，例如多层感知机（MLP），尝试拟合乘法器，并展示结果。本次作业需要大家自行生成数据集，例如：随机生成5000组随机数（a, b），对应的label即为a \* b，其中4000组数据作为训练集，1000组数据作为测试集。

将你的结果和你对这个问题的实现与理解写成一页pdf的形式。提交作业时上传全部的python代码，与pdf报告。在DDL前发送至canvas上。（报告限制pdf一页）

**第二次小作业：CNN初探（15分）**

目标：

1. 下载MNIST数据集
2. 搭建国产深度学习训练框架Jittor，Mindspore 或 Paddle （三选一，推荐Jittor），并设计CNN模型
3. 在给定的要求下改进模型

MNIST数据集下载：<http://yann.lecun.com/exdb/mnist/>

任务：

在国产深度学习框架Jittor，Mindspore 或 Paddle （三选一，推荐Jittor）中，利用神经网络解决MNIST手写数字识别这一任务。由于MNIST数据集比较经典，因此我们加了额外要求。

要求：

1. 构建神经网络在MNIST数据集上训练测试，并汇报结果。
2. 将训练集进行划分，具体做法为所有（0, 1, 2, 3, 4）的图像仅保留10%，剩余部分不变。
3. 在新的训练集上训练神经网络，这时结果相较于原先会有下降，那么，提出可能的解决方案，尝试改进你的模型。
4. 将你在新训练集上的分析，模型设计改进，连同实验结果一起，写成两页pdf报告的形式，在DDL前与代码一起上传到Canvas上。（报告限制pdf两页）

训练集切分方式示例：

if i % 10 != 0:

mask = (labels >= 5)

else:

mask = (labels >= 0)

labels = labels[mask]

images = images[mask]

**第三次小作业：RNN初探（15分）**

目标：

1. 下载MNIST数据集

1. 搭建国产深度学习训练框架Jittor，Mindspore 或 Paddle （三选一，推荐Jittor），并设计RNN模型
2. 在给定的要求下改进模型

MNIST数据集下载：<http://yann.lecun.com/exdb/mnist/>

任务：

上一个作业我们尝试了使用CNN来解决MNIST手写数字识别这一任务，那么接下来，我们尝试使用RNN来再次解决这一问题。用RNN解决分类问题的思路有很多，最直观的方法是：将图片（28\*28）视为由28条一维数据组成的，通过RNN网络学习这28行数据，来输出一个分类，并与label相比较来计算损失函数并更新神经网络。同学们如果有自己的设计也欢迎尝试。本次作业的任务目标与上一次作业相同，只是将CNN网络换成了RNN网络，大家在完成作业的过程中也可以思考一下CNN与RNN的差异与特性。

要求：

1. 构建神经网络在MNIST数据集上训练测试，并汇报结果。

1. 将训练集进行划分，具体做法为所有（0, 1, 2, 3, 4）的图像仅保留10%，剩余部分不变。
2. 在新的训练集上训练神经网络，这时结果相较于原先会有下降，那么，提出可能的解决方案，尝试改进你的模型。
3. 将你在新训练集上的分析，模型设计改进，使用CNN和RNN网络的异同与感受，连同实验结果一起，写成三页pdf报告的形式，在DDL前与代码一起上传到Canvas上。（报告限制pdf三页）

训练集切分方式示例：

if i % 10 != 0:

mask = (labels >= 5)

else:

mask = (labels >= 0)

labels = labels[mask]

images = images[mask]

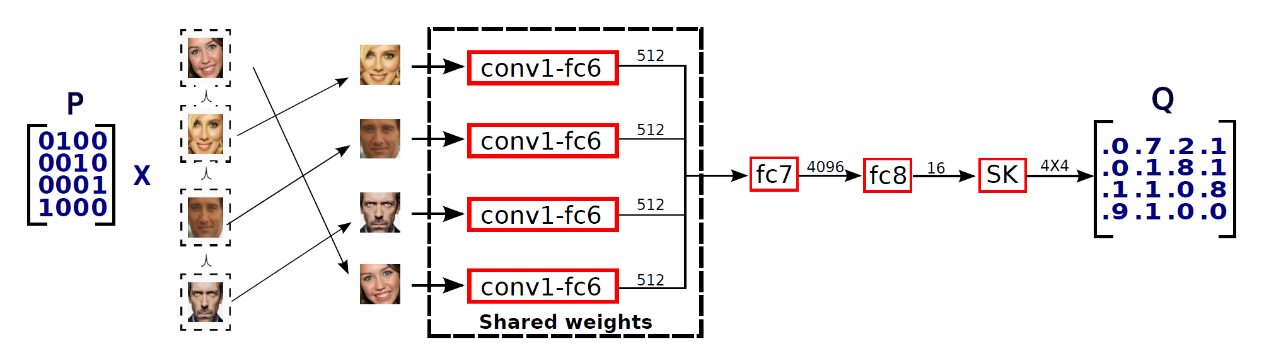
* 大作业（60分）

本次深度学习课程大作业基于“视觉”与“图”两大深度学习的重点研究方向，对于一个视觉问题中的应用（拼图问题），在参考已有论文的基础上，提出自己的解决方案。

目标：

在CIFAR-10数据集上，我们将在图像分类任务以外研究图片拼接这一任务，即，对于一张图片，我们将其切分为若干张子图片，例如水平竖直分为四份。接着，我们将这几张子图片打乱顺序作为输入，目标就是设计一个模型可以根据这些乱序的子图片还原出他们本来的排列位置。

相关参考论文：DeepPermNet: Visual Permutation Learning <https://arxiv.org/abs/1704.02729>



任务步骤：（这里只是推荐的实现步骤，如果有不同的设计也可以自行实现，最终我们将依据pdf报告中展示的完成质量综合考虑进行打分）

1. 修改作业二/作业三中的图片分类的代码框架，将输入图片分割为N张子图片，打乱它们的顺序作为输入，并将他们的正确排列作为label，输出格式采用排列阵P，其中第i行第j列为1则代表第i个子图片应该排在第j个位置。注意，每行与每列均最多有一个元素为1。具体切分方式可以自行定义。
2. 设计一个针对图片拼接的神经网络模型，可以参考DeepPermNet的设计，即将每一个子图片输入一个CNN进行特征提取，将得到的特征进行拼接后通过若干全连接网络得到一个N x N 的矩阵，使用Sinkhorn算法使其变成双随机矩阵Q（即每行每列的和固定为1）作为神经网络的预测值。最后将预测值Q（双随机矩阵）与真值P（排列阵）作为损失函数的输入，计算Loss后用于更新神经网络。
3. 神经网络可以套用作业二/作业三中的代码，也可以自行设计新的网络。Sinkhorn算法推荐使用pygmtools中的[Sinkhorn](https://pygmtools.readthedocs.io/en/latest/api/_autosummary/pygmtools.linear_solvers.sinkhorn.html)，这是我们实验室自行研发的开源平台。

本作业侧重探索性质，对精度不做要求，没有固定的解决方案，言之有理即可。最终要求提交一份五页pdf报告，在DDL前与代码一起上传到Canvas上。（报告限制pdf五页）