
作业 3：深度学习

清华大学软件学院
人工智能导论, 2025 年春季学期

介绍

本次作业需要提交说明文档（PDF 形式）和 Python 的源代码。注意事项如下：

- 本次作业满分为 100 分，附加题 5 分，得分超过 100 分按 100 分记。
- 除简答题、编程题外的题目，请给出必要的解答过程，只有答案且过于简略的回答会酌情扣除一定分数。题目要求直接回答或只汇报结果的题目，则不需要给出过程或分析。
- **请不要使用他人的作业，也不要向他人公开自己的作业，复制网上内容须在报告中说明**，否则将受到严厉处罚，作业分数扣至-100（即倒扣本次作业的全部分值）。
- 完成作业过程中，如果使用了大模型辅助（如润色文笔、询问知识点等），请在作业末尾声明使用的方式和程度（不影响作业评分）。**禁止直接粘贴大模型输出的文本**，否则会扣除一定的作业分数。
- 统一文件的命名：{学号}_{姓名}_hw3.zip。所有解答和实验报告请写在一个 pdf 文件中，和代码一起压缩上传。

1 简答题（15 分）

1. 什么是交叉熵（Cross Entropy）？在学习一个类别分布（Categorical Distribution）时，使用交叉熵作为损失函数比绝对值损失函数（Absolute Error, $L_{abs} = |y_i - \hat{y}_i|$ ）有什么好处？
2. 多层感知机（Multilayer Perceptron）相比线性模型有哪些优势？相较于训练浅而宽的神经网络（“宽度学习”），训练相对窄而深的神经网络有什么好处？
3. 卷积（Convolution）和互相关（Cross-correlation）分别是什么意思？在卷积神经网络中，卷积核通常进行的是卷积还是互相关操作？
4. 批量大小（Batch Size）对于优化器（比如随机梯度下降）影响巨大。为了减小内存占用，小宣提出将每次前向传播的批量大小减半，梯度累积两次再进行反向传播。请问这种方法能确保训练得到的模型效果参数一致吗（假设随机状态、batch 划分、dropout 的神经元相同）？若有影响，请指出原因（例如优化器、模型中的某些层）；若无影响，请论证。
5. 为什么说残差连接（Residual Connection）有利于训练更深层的深度网络？残差链接能够缓

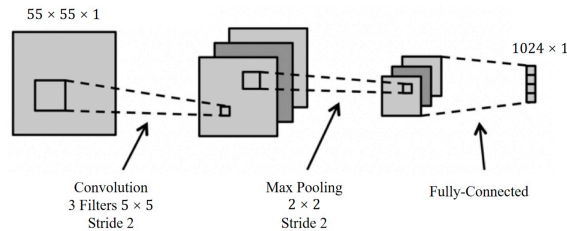
解梯度消失 (Gradient Vanishing) 的问题吗?

2 解答题

2.1 卷积神经网络 (10 分)

一个卷积神经网络的前向传播过程如下图所示, 它依次通过以下各层将一张尺寸为 $55 \times 55 \times 1$ 的单通道图片转换为 1024×1 的输出: 卷积层 (Convolution)、最大池化层 (Max Pooling)、ReLU、全连接层 (各层参数已在图中列示), 且在该网络中, 我们将不使用任何偏置参数 (Bias Parameters)。则对于该网络:

1. 卷积层共有多少个可学习参数?
2. 最大池化层的输出的尺寸为多少?
3. 在前向传播过程中, 对于每个样本需要进行多少次 ReLU 函数计算
4. 为了给模型加入非线性, 需要在网络中加入激活函数, 请列举两个激活函数。



2.2 注意力机制 (25 分)

本题中我们将探究利用 GPT 类架构进行机器翻译过程中的注意力机制计算过程。

假设我们想要翻译“他 | 喜欢 | 苹果”这一中文句子 (3 个 token 使用竖线分隔), 在 GPT 的某自注意力层中 3 个 token 的 query、key、value 向量分别记作 $Q = \{q_1, \dots, q_3\}$, $K = \{k_1, \dots, k_3\}$, $V = \{v_1, \dots, v_3\}$, $q_i, k_i, v_i \in \mathbf{R}^d$ 。

1. 请写出经过带掩码的自注意力层

$$Y = \text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V$$

之后每个 token 对应的输出 y_i 的表达式。(注意以上公式中省略了掩码)

2. 设 $d = 4$, $K = \left\{ \begin{bmatrix} 0 \\ 1 \\ -1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ -1 \\ 0 \\ -1 \end{bmatrix} \right\}$ 。当 GPT 模型预测翻译的第一个 token (英文单词)

“He” 的时候, 它应该需要尽量多的来自 token “他” 的信息。请写出向量 q_3 的一个取值, 满足 q_3 的 2 范数不超过 1, 且与第一个 token 的自注意力权重 (相比别的 token) 最大, 并写出此时 y_3 关于 v_1, v_2, v_3 的表达式。

3. 设生成了 token “He” 之后, 计算到这一层时, $k_4 = (-1, 0, -1, -1)^T$ 。当 GPT 模型预测翻译的第二个 token (英文单词) “likes” 的时候, 它同时需要 “喜欢” 和 “He” 的信息 (因为

“likes” 是第三人称单数形式), 此时多头自注意力机制可以胜任。假设以 query, key, value 向量的前两维和后两维作为两个自注意力头 (Heads) 的特征向量, 请写出向量 q_4 的一个取值, 满足 q_4 的 2 范数不超过 1, 且在第一个自注意力头中与第二个 token (“喜欢”) 的自注意力权重 (相比别的 token) 最大, 在第二个自注意力头中与第四个 token (“He”) 的自注意力权重最大, 并写出此时 y_4 的表达式。

2.3 [附加题] 感知机的收敛保证 (5pt)

在课程中, 我们讲解了感知机 (Perceptron), 一个经典的学习算法。下面, 让我们考虑感知机对于 -1,1 分类任务下的收敛保障。具体来讲, 在这种设定下, 我们有数据集 $\mathcal{D} = \{(x, y), x \in \mathbb{R}^d, \|x\| < 1, y \in \pm 1\}$, 我们希望学习一组参数 $w \in \mathbb{R}^d$, 对于数据 x 的标签 y 进行预测: $f(x) = \text{sign}(w^\top x)$ 。在此设定下, 感知机算法如下:

Algorithm 1: Perceptron Learning Algorithm

```
1:  $P \leftarrow$  inputs with label 1
2:  $N \leftarrow$  inputs with label -1
3: Initialize  $w = 0$ 
4: while not Convergence do
5:   Pick random  $x \in P \cup N$ 
6:   If  $x \in P$  and  $w^\top x < 0$  then
7:      $w = w + x$ 
8:   end if
9:   If  $x \in N$  and  $w^\top x \geq 0$  then
10:     $w = w - x$ 
11:  end if
12: end while
```

下面, 请证明:

若有 $\forall i \in |\mathcal{D}|, \|x_i\| < 1, \exists w^* \in \mathbb{R}^d, \gamma > 0, s.t. \|w^*\| = 1, \forall (x_i, y_i) \in \mathcal{D}, y_i w^{*\top} x_i > \gamma$ (即存在一个过原点的划分平面, 有安全距离 γ)。该算法收敛前最多触发 $\frac{1}{\gamma^2}$ 次预测错误。

3 深度学习与 AlphaZero (50 分)

问题背景 在之前的作业中, 我们实现了 AlphaZero 的训练流程, 并尝试用线性模型学习围棋的策略和价值函数。但是, 简单的线性模型并不足以建模复杂的围棋问题, 我们需要引入深度神经网络来解决围棋问题。

任务目标 本次作业中, 我们将继续在 7×7 围棋问题上探索 AlphaZero 算法的能力。下发的代码文件中, 已经实现了一个简单的全连接网络的示例 (model/example_net.py:MLPNet, 你需要先利用示例网络运行 AlphaZero 训练, 然后参考示例网络, 自己设计并实现一个深度网络, 测试训练效果。提交时请删除 *.so、*.pyd 和 */build/等临时文件和训练过程的 checkpoint, 仅提交代码和 1 个最好的模型参数文件, 本题的文字报告请和其他题目写在同一个文档中提交。

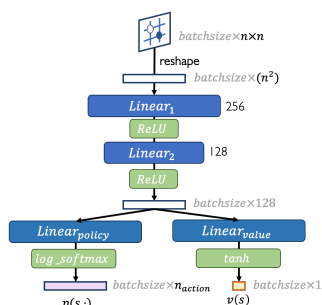
1. 将之前作业中完成的代码填入对应位置，运行训练，并汇报使用 MLP 模型的 AlphaZero 算法训练过程中对 Random Player 的胜率，和训练过程的 elo 分数曲线图。
2. 参考示例代码，设计并实现一个不一样的深度模型，要求至少需要使用一个卷积层处理二维棋盘特征。请绘制网络结构图，并简要说明设计的理由。
3. 使用自己设计的深度模型，运行训练，并汇报训练过程中 AlphaZero 算法对 Random Player 的胜率以及 elo 分数曲线图。要求训练过程中，对 *Random Player* 的胜率至少有一次不低于 90%。
4. 修改 `pit_puct_mcts.py`，加载训练后的 MLP 模型和自己设计的模型进行对弈，汇报对局的胜率。

参数选择 完成上述题目时，可以自由选择适合你的情况的参数进行训练，但过于不合理的参数设置可能会导致扣分（若报告未说明实验使用的参数，则会以提交的代码为准）。

文件大小限制 提交的模型参数文件大小不能大于 **32MB**，且只能提交 1 个你认为效果最好的模型参数文件（有特殊情况请与助教提前沟通）。

提示

- 在 `model/example_net.py` 中预留了一个 `MyNet` 类用于实现你自己设计的深度模型。你也可以将其重命名为合适的名字。
- 模型设计不是越复杂越好，过于复杂或者参数量过大的模型可能导致训练缓慢、容易过拟合。
- 卷积层的实现你可能会用到 `torch.nn.Conv2D1` 和 `torch.nn.BatchNorm2D2`。
- 推荐使用并行脚本（`alphazero_parallel.py`）进行训练，并根据实验环境实际情况，使用合适并行数（主函数中 `N_WORKER` 变量控制）。训练时间可能较长，建议提前评估合理安排。
- 本次作业下发的训练脚本默认会覆盖同一保存路径下的文件，训练时请做好备份，或确保使用了不同的保存路径。
- 请确保前两次作业涉及的代码实现正确。尽管前两次作业的内容不在本次作业考察的范围内，但是错误的实现可能导致异常的训练结果。你可以参考已发布的优秀作业，检查自己代码中的问题。
- 示例的 MLP 模型的网络结构图参考：



¹<https://pytorch.org/docs/stable/generated/torch.nn.Conv2d.html>

²<https://pytorch.org/docs/stable/generated/torch.nn.BatchNorm2d.html>

4 提交格式

- 请先删除 *.so *.pyd 等文件和环境目录下的 build 文件夹，再将你的代码目录内**所有代码文件**、**1 个最好的模型参数文件**和你的**文字报告**打包提交。

统一文件的命名：{学号}_{姓名}_hw3.zip。

- 请将本次作业所有问题回答写在同一份报告中，报告请导出为 **pdf 格式**。