

摘要

尝试使用神经网络的深度学习较为快速和便捷的解决糖尿病眼底图像的分类任务。使用了 convmixer 模型进行学习，并取得了较好的成果。

目录

1 前言

2 代码思路

2.1 基本思路

2.2 数据处理

2.3 梯度下降算法

3 实现过程

3.1 简单的卷积神经网络

3.2 resnet 网络

3.3 convmixer 网络

4 评价标准及试验结果与分析

5 总结

1 前言

在如今的生活中，个人的生活水平不断的提高，人民的生活逐渐富足；但是在生活质量逐渐发展的同时，各种疾病的患病人数和亚健康状态人数不断增加，糖尿病是其中比较严重、且大众关注度最高的一种疾病。糖尿病的判断可以通过眼底图像的病变严重程度进行判断，其核心是对于眼底图像进行分类，因此尝试使用神经网络深度学习进行图形分类任务对此项工作进行解决。

在设计过程中使用了 DDR 数据集，其中包括 train、valid、test 三个数据集，每个数据集中均包括从正常病人（0 类）到较为严重的病人（4 类）共五类分类。首先将使用 train 数据集中的数据进行训练，并使用 valid 数据集中的数据进行验证并调试超参数，最后使用 test 数据集进行测试。

2 代码思路

2.1 基本思路

对于本次任务，基本思路是使用卷积神经网络进行图像分类任务。

上图显示了全连接神经网络和卷积神经网络的区别。全连接神经网络是一种模仿动物神经网络行为特征，进行分

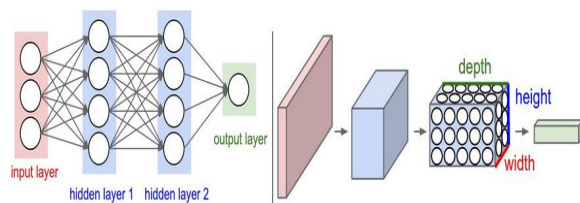


图 2.1. 卷积神经网络与全连接神经网络的区别

布式并行信息处理的算法数学模型，由大量神经元相互交织构成，其最后的输出结果取决于由神经元之间的连接等的不同；卷积神经网络通过模拟人脑来进行数据解释，通过一层层的卷积对于传入的数据进行特征提取，将输入数据的特征逐渐暴露并强化这些特征，将这些暴露的特征进行记忆，为下一次任务做好准备。

本次任务正是基于卷积神经网络进行图像分类任务。首先进行模型构建，设置卷积层的层数、输入输出的通道数、池化层、全连接层以及训练次数、batchsize 等超参数；然后进行准确率函数的构建，使用混淆矩阵，并通过交叉熵对结果进行准确率分析；最后对模型进行训练和测试。

2.2 数据处理

本次任务中需要进行卷积操作，为了使得图像可以进行卷积操作，需要首先对图像进行处理，将图像转换成 dataloader 的格式。第一步使用 datasets.ImageFolder() 函数将图像进行装载，同时由于完整的图像过大，容易导致显存爆炸，因此在这一步对图像进行切割，由 512*512 大小重新处理为 256*256 大小；第二步使用 torch.utils.data.DataLoader() 函数将装载好的数据变成 dataloader 的格式，同时将数据分割成一个个 batch，并同意进行数据打乱。

2.3 梯度下降算法

神经网络深度学习任务的核心是梯度下降算法。梯度下降就是求一个函数的最小值，对应的梯度上升就是求函数最大值。简而言之：梯度下降的目的就是求函数的极小值点，例如在最小化损失函数或是线性回归学习中都要用到梯度下降算法。

$$L(w, b) = \frac{1}{N} \sum_{i=1}^N (y_i - f(wx_i + b))^2$$

图 2.2. 梯度下降的数学意义

通过梯度下降算法方可得出模型一次学习后的结果是好是坏，通过一次一次的循环学习最后能够到达一个较好的结果。

3 实现过程

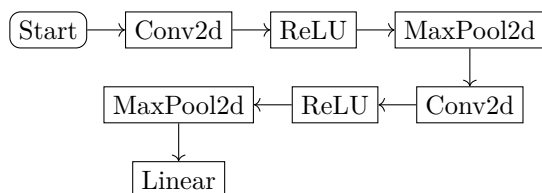
在本次任务中尝试了很多网络模型，包括简单的卷积神经网络、resnet 网络和 convmixer 网络。在尝试过程中添加了一些可视化工具，包括 tensorboard、训练过程进度条显示、混淆矩阵的热力图实现等。具体的学习数据如下：

模型	训练集准确率	测试集准确率
CNN	98%	52%
resnet	100.0%	60.9%
convmixer	93.61%	76.24%

表 3.1. 使用的模型及训练集、测试集正确率

下面将具体一一说明。

3.1 简单的卷积神经网络



本模型使用了两个卷积层，两个 relu 层，两个池化层和一个全联接层，最终训练结果为：训练集上准确率 98%，测试集上准确率 52%。

在使用模型时选择将一个卷积层、一个 relu 激活层、一个池化层组成一个套餐，方便模型的构建。通过反向传播实现梯度下降，并使用了 adam 优化器对学习率进行管理。

在进行后续分析时发现测试集上的准确率显著低于训练集上的准确率，出现了欠拟合现象，分析原因发现是由于网络深度不够，没有学到太好的特征所致。后续对模型进行了修改，添加了两个套餐，发现测试集准确率有一定提升。

问题： 当继续在原先的模型基础上进行添加套餐时出现了测试集准确率反而下降的现象，分析是由于梯度消失导致的过拟合现象。

3.2 RESNET 网络

由于在上述网络中出现了过拟合现象，考虑残差网络 (resnet) 对梯度消失和梯度爆炸问题进行规避。

resnet 的思想是经过一次运算后梯度比原先的要大，此时为了使结果至少不比上一层运算结果差就直接将上一层的运算结果作为下一层的输入。

在构建模型时，首先构建 residual 类，目的是构建残差结构；之后定义自己的网络类，在每一个套餐内使用残差结构预防梯度爆炸或梯度消失；最后进行调用学习即可。

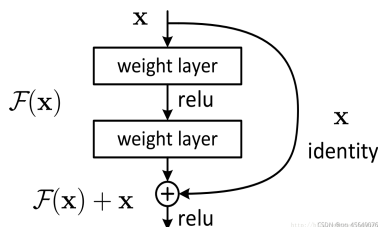


图 3.1. resnet 网络

在进行学习后，损失：0.007205，训练集准确率：100.00%，测试集正确率：60.39%。尽管已经较之一般的 CNN 网络有了较大的进展，但是并不是 resnet 应有的水准，因此进行分析：第一，网络深度不够，本身 resnet 的出现就是为了解决网络深度无法增加的问题，依旧使用浅网络自然结果一般；第二，有一些超参数并未调试好。

首先是对模型进行加深，然而出现问题：当模型加深到 40 层左右后，模型容易出现自身过大而导致电脑显存爆炸的问题，因此继续加深网络是不可取的。然后使用 valid 集对于超参数进行调整：首先扩大了 batchsize，这使得模型能在一轮学习过程中学到更多的数据；其次是传的的图片随机裁剪为更大的图片，可以使模型学得更多的信息，但全图片 (512*512) 大小太大，容易出现内存爆炸，最后权衡使用 256*256 大小的随机裁剪图片。

问题： 模型过于庞大，学习速度很慢，不容易对模型进行修改。

3.3 CONVMIXER 网络

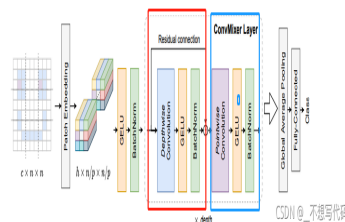


图 3.2. convmixer 网络

经过查询发现了一个较新的、基于残差网络构建的网络模型：convmixer。ConvMixer，是把通过卷积映射成一个一个的特征块输入到网络中。网络也不会通过下采样（池化）来改变输出的维度。整个网络结构通过传统的卷积来实现。根据论文所述，当深度可分离卷积部分的卷积核越大，模型的性能越好。文章最后也认为，ViT 表现如此优越是因为 patch embedding（图片分块）的原因。作者认为 patch embedding 操作就能完成神经网络的所有下采样过程，降低了图片的分辨率，增加了感受野，更容易找到远处的空间信息。

实现 convmixer 时，首先定义了 ConvMixerLayer 类，其中包括一个残差结构套餐和一个逐点卷积套餐。之后定义 ConvMixer 类，目的是生成所需要的模型。

convmixer 的优势明显：第一，其拥有残差结构，因

此不必担心梯度爆炸或者梯度消失的问题；第二，其整体复现较为容易；第三，整体深度不高，不容易出现显存爆炸的危险。

最初的 convmixer 模型准确率约为 70%，通过加入数据增强、预训练、调参准确率提升至 76.24%。

4 评价标准及试验结果与分析

本次试验使用了交叉熵和混淆矩阵来进行准确率判断。交叉熵主要是用来判定实际的输出与期望的输出的接近程度；混淆矩阵 (Confusion Matrix)，混淆矩阵是可视化工具，用于比较分类结果和实际测得值，可以把分类结果的精度显示在一个混淆矩阵里面。在本次试验中，通过交叉熵得到准确率，通过混淆矩阵进行可视化验证。

最后使用 convmixer 训练 98 轮的结果如下：98 轮训练后，loss (损失值) 为 0.240865，训练集上准确率为 93.61%，测试集上准确率为 76.24%，验证集上准确率为 83.46%。下面是 tensorboard 对整个训练过程中 loss、train 准确率、test 准确率的图像拟合：

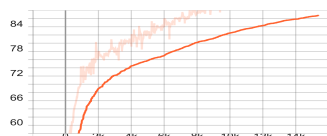


图 4.1. train 拟合结果

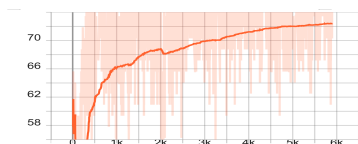


图 4.2. test 拟合结果

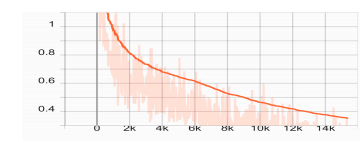


图 4.3. loss 拟合结果

在最后一轮训练后得出的混淆矩阵为：

```
[[1784, 103, 384, 1, 8,],
 [28, 14, 27, 0, 3,],
 [67, 69, 884, 44, 96,],
 [1, 2, 19, 22, 6,],
 [0, 1, 30, 4, 162,]]
```

混淆矩阵可视化：

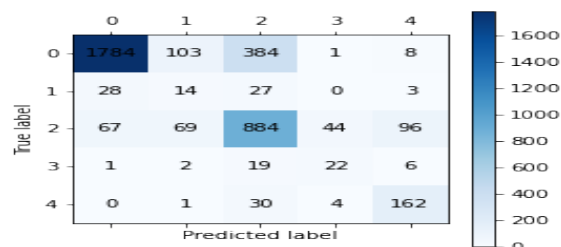


图 4.4. 混淆矩阵

通过上述混淆矩阵可以发现，由于 1 类的特殊性（得病特征不明显），很容易被误分类为 0 类和 2 类；其余的分类结果良好。

5 总结

本次任务使用了 convmixer 对糖尿病病人眼底图像进行分类任务，结果较好。在模型的构建过程中使用了 tensorboard 可视化工具、rich 可视化进度条包、混淆矩阵等一系列可视化工具；使用了预训练、图像增强等数据加强手段。学习了很多，收获满满。