

深度学习实验三

1190200708 熊峰

1190200704 管建男

2022 年 6 月 8 日

目录

一 环境配置	3
1.1 硬件配置	3
1.2 软件配置	3
二 模型结构	3
2.1 VGG	3
2.2 ResNet	4
2.3 SENet	5
三 优化方法	6
3.1 公共优化部分	6
3.1.1 优化器	6
3.1.2 数据增强	6
3.2 其他优化方法	6
四 训练过程	6
4.1 无数据增强的情况	6
4.2 有数据增强的情况	7
4.3 进一步增强：使用 K 折交叉验证的 Swin Transformer 模型	9
五 实验结果	9
六 成员分工	9
6.1 熊峰	9
6.2 管健男	10

一 环境配置

1.1 硬件配置

CPU : Intel(R) Xeon(R) Silver 4214

GPU : TITAN RTX 24G

MEM : 128G RAM

1.2 软件配置

OS : Ubuntu 20.04.1 LTS

PyTorch : Stable 1.11.0 CUDA 11.3

IDE : PyCharm 2021.3.2

二 模型结构

2.1 VGG

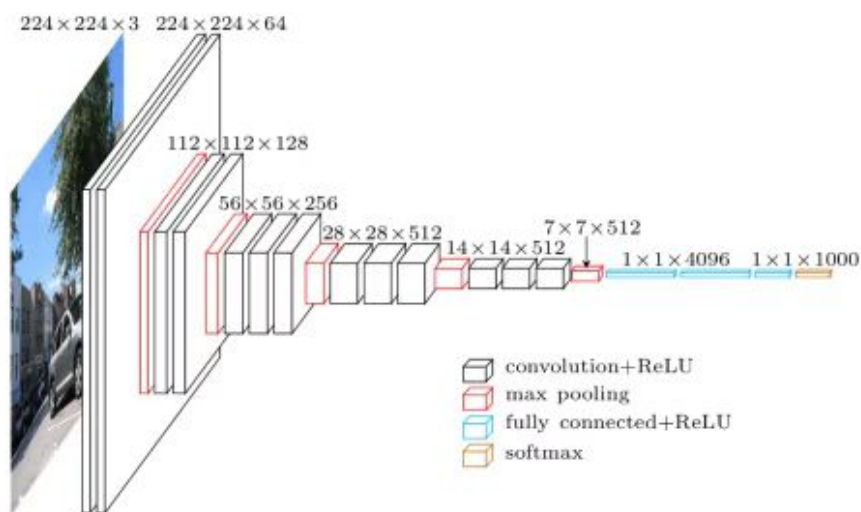


图 1: VGG16 Model

本部分主要实现了 VGG16 模型 [1]。

第一部分经过两层卷积 (在所有卷积操作后加上 ReLU), 此时输出为 $64 \times 224 \times 224$.

第二部分经过一层池化层, 再通过两层卷积层, 此时输出为 $256 \times 56 \times 56$.

第三部分经过一层池化层, 再通过三层卷积层, 此时输出为 $512 \times 14 \times 14$.

第四部分经过一层池化层，此时输出为 $512*7*7$ 。

第五部分主要为全连接层，将上一部分输出映射为 12 维的向量。

2.2 ResNet

深的神经网络难以训练，何恺明提出残差学习 [2] 的框架，使训练非常深的网络比之前容易很多。其将各层重新表述为学习参考层输入的残差函数。

当网络很深时，容易出现梯度消失、梯度爆炸等问题，通常采用初始化的权重和在中间加入 Normalization，例如 BN 等，可以校验每个层之间的输出和他的梯度的均值和方差。使用上述方法可以使网络收敛，这时一个次要的问题就暴露出来了。随着网络的深度增加，准确度会达到饱和，然后迅速退化，这种退化并不是由于过拟合引起的，而在一个适当的深度模型上增加更多的层会导致更高的训练误差。

因此 ResNet 中提出了残差学习框架，其不希望几个堆积层直接训练所需的映射，而是让这些层训练一个残差的映射 $\mathcal{F}(x) = \mathcal{H}(x) - x$ 。

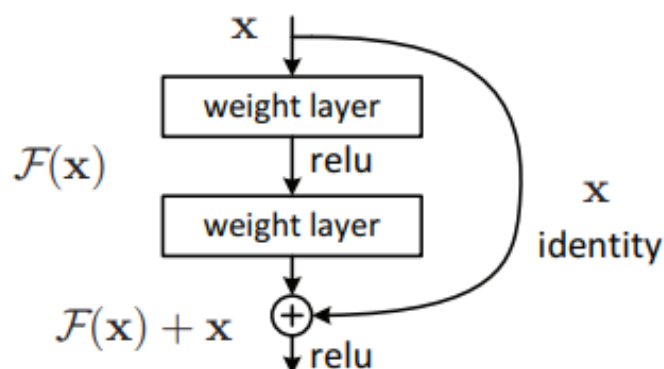


Figure 2. Residual learning: a building block.

图 2: ResNet Block

x 为浅层网络学习到表征，经过堆叠的层后得到的映射为 $\mathcal{H}(x)$ ，但不让这个堆叠的网络直接训练得到 $\mathcal{H}(x)$ ，因为直接学习到的很有可能效果不如浅层网络，因此希望它学习 $\mathcal{F}(x) = \mathcal{H}(x) - x$ ，这个函数更易于优化，假定此时恒等映射更好，即 $\mathcal{H}(x) = x$ ，那么可以使堆叠的网络学习到的为 $\mathcal{F}(x) \rightarrow 0$ ，即将残差部分推到 0，此时 $\mathcal{H}(x) = x$ ，此时至少可以保证效果不低于浅层网络，若堆叠的层获得了更好的表征，很有可能提升网络的性能。

2.3 SENet

SENet[3]是 ImageNet 2017 的冠军模型，其主要思想为，对于每个输出 channel，预测一个常数权重，对每个 channel 加权。其可以很方便地集成到现有网络中，提升网络性能，并且代价很小。

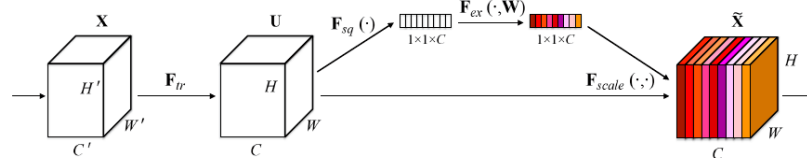


图 3: SENet

如图为，SENet 和 ResNet 的结合，第一层的 FC 会把通道降下来，然后第二层 FC 再把通道升上去，得到和通道数相同的 C 个权重，每个权重用于给对应的一个通道进行加权。上图中的 r 就是缩减系数，实验确定选取 16，可以得到较好的性能并且计算量相对较小。

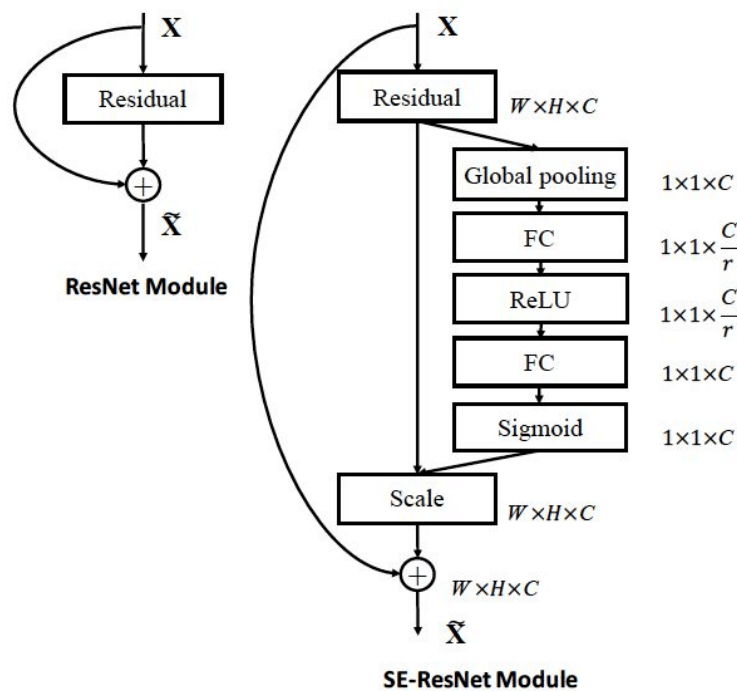


图 4: SENet 和 ResNet 的结合

三 优化方法

3.1 公共优化部分

3.1.1 优化器

Adam: 使用 Adam 训练效果较好, 在较多模型都得到比较好的结果。

SGD: 使用 SGD 训练时, 大多数模型效果很差, Micro F1 Score 维持在 0.1-0.2.

3.1.2 数据增强

进行 data augmentation (翻转、旋转、移位等操作) 对比。

3.2 其他优化方法

引入新的模块/换用更好的模型等内容不限, 可有效提升性能即可
增加一种 Swin Transformer 模型 [4]。

使用 K 折交叉验证。

四 训练过程

4.1 无数据增强的情况

VGG 训练过程中的 micro F1 和 macro F1 如图5

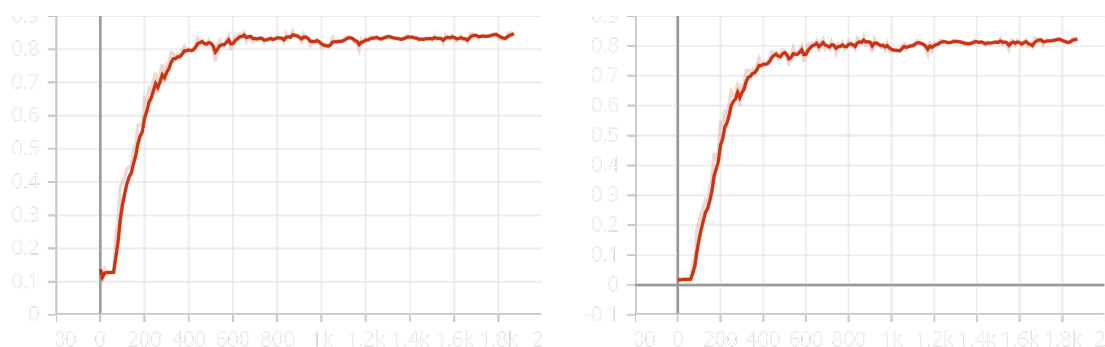


图 5: VGG 训练过程中的 micro F1 和 macro F1

ResNet 训练过程中的 micro F1 和 macro F1 如图6

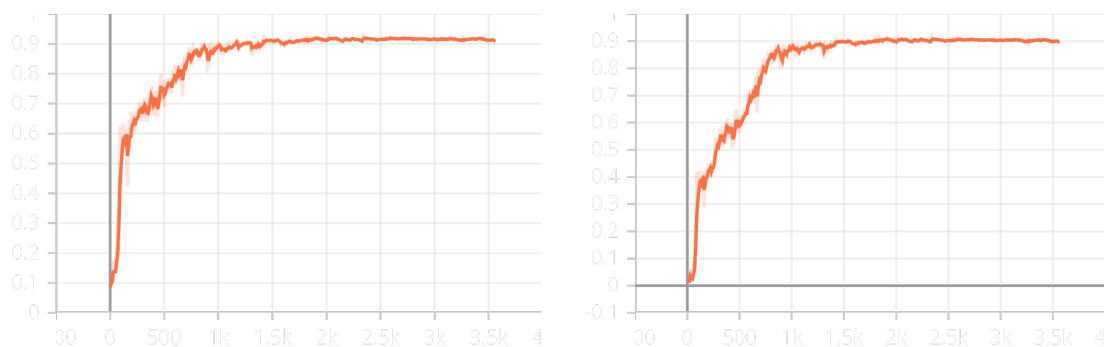


图 6: ResNet 训练过程中的 micro F1 和 macro F1

SENet 训练过程中的 micro F1 和 macro F1 如图7

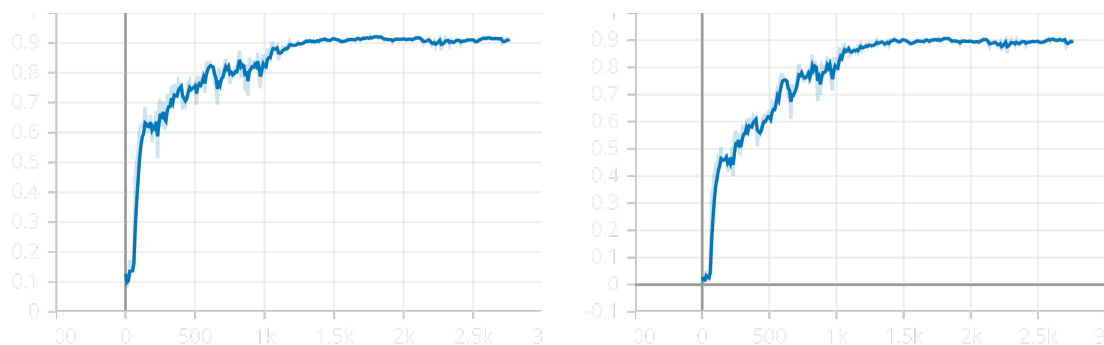


图 7: SENet 训练过程中的 micro F1 和 macro F1

4.2 有数据增强的情况

采用的数据增强包括旋转 (rot)、翻转 (flp)、色彩增强, 其中色彩增强又包括分色 (pst) 和曝光 (slr)。数据增强的具体操作方式为: 在原始数据集的基础上, 通过水平翻转 (hflp) 和垂直翻转 (vflp) 产生四份新的数据集, 再对每一份数据集做色彩增强。

采用了全部的数据增强方式, VGG 训练过程中的 micro F1 和 macro F1 如图8

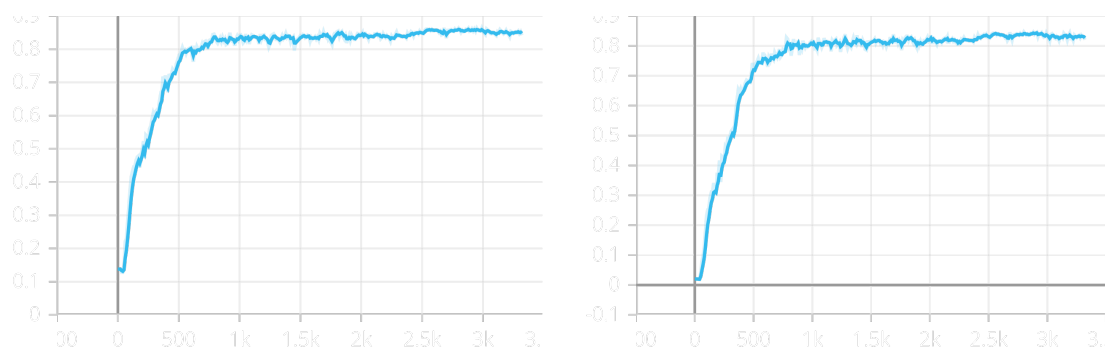


图 8: 采用了全部的数据增强方式, VGG 训练过程中的 micro F1 和 macro F1

采用了全部的数据增强方式, ResNet 训练过程中的 micro F1 和 macro F1 如图9

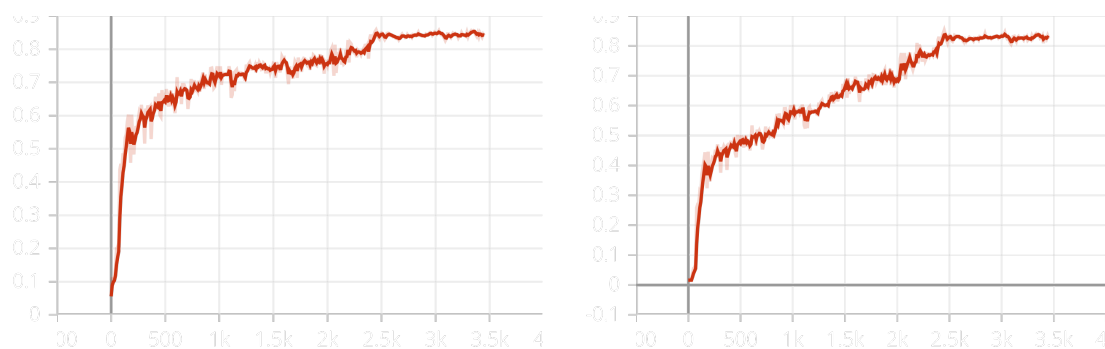


图 9: 采用了全部的数据增强方式, ResNet 训练过程中的 micro F1 和 macro F1

采用了全部的数据增强方式, SENet 训练过程中的 micro F1 和 macro F1 如图10

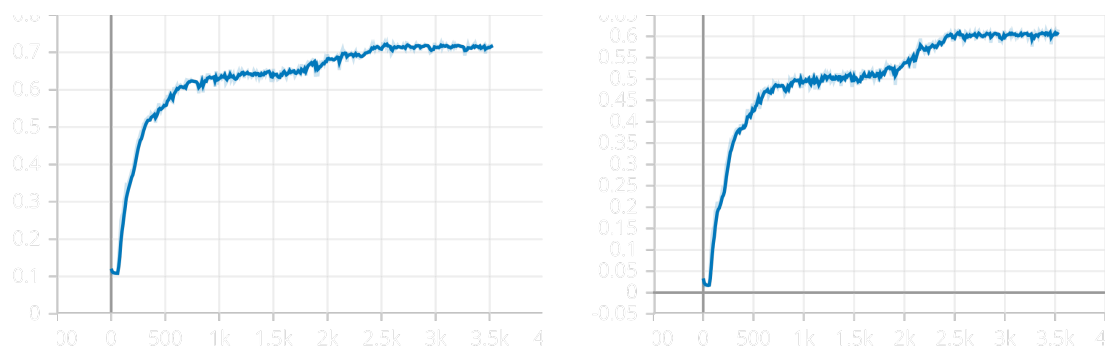


图 10: 采用了全部的数据增强方式, SENet 训练过程中的 micro F1 和 macro F1

4.3 进一步增强：使用 K 折交叉验证的 Swin Transformer 模型

使用 K 折交叉验证的 Swin Transformer，训练过程中的 micro F1 和 macro F1 如图11
由于实验中 K 取值为 5，因此下图中展示了这 5 个模型的结果。

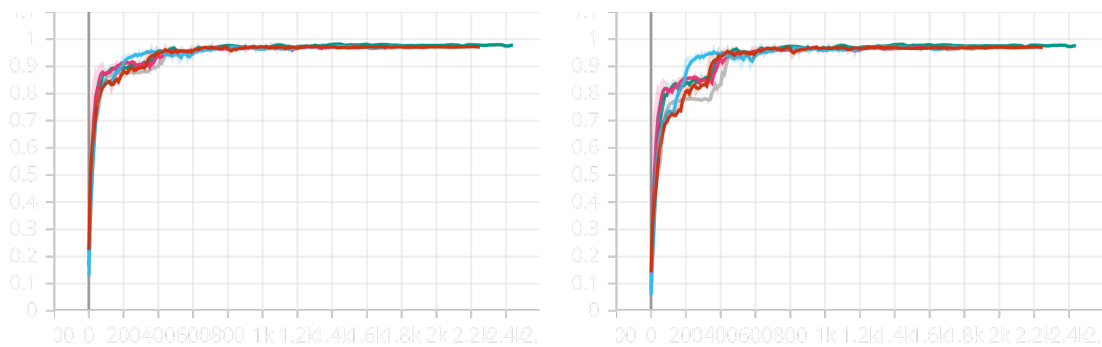


图 11: Swin Transformer 训练过程中的 micro F1 和 macro F1

五 实验结果

各个模型实验结果如图五

Model	Augmentation	Micro F1	Macro F1	Kaggle
VGG16	None	85.05%	82.76%	85.39%
VGG16	vflip+hflip+pst+slr	84.21%	82.08%	85.77%
ResNet18	None	92.84%	91.76%	92.19%
ResNet18	vflip+hflip+pst+slr	82.44%	82.44%	85.89%
SENet18	None	92.21%	90.96%	91.94%
SENet18	vflip+hflip+pst+slr	72.74%	61.61%	62.97%
Swin Transformer	feature extractor + Kfold	97.28%	97.06%	97.48%

表 1: 各个模型实验结果

六 成员分工

6.1 熊峰

1. 实现 VGG、ResNet、SENet 等模型，个人优化使用 Swin Transformer 模型。
2. 编写训练过程
 - 编写基础训练过程

- 个人优化: 训练过程增加 K 折交叉验证
- 3. 训练 SENet、更新模型使用 Swin Transformer 模型
- 4. 撰写报告模型部分

6.2 管健男

1. 实现 argparse 模块
2. 实现支持 K 折交叉验证的数据加载器
3. 实现数据增强模块
4. 训练 VGG、ResNet 模型
5. 撰写报告训练结果部分

参考文献

- [1] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2.1
- [2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2.2
- [3] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 2.3
- [4] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 3.2