REPORT OF AI PROJECT 2

Build a Simple Classifier: From CNNs to ViTs

Author
李逸飞
Dec 21st, 2023

Contents

1 Introduction

本文是人工智能项目 2 的技术报告。本任务要求学生构建一个简单的分类器，分别适用于二分类、五分类以及多标签分类。为完成任务要求，笔者自己构建了一套完整的数据集处理、模型构建、训练和评估的框架。我们认为该框架简介易用，同时可以迅速加以少量的改变以迁移到其他任务中去。除此之外，笔者还在二分类和无分类任务中分别使用了四种 CNN 的框架以及一种 ViT 的框架，并对上述五种框架的表现进行了对比。

在本报告中，笔者将首先介绍所使用模型的基本原理，包括四种基于 CNN 的方法：AlexNet[1], VGG[2], ResNet[3]，以及一种基于 Vision Transformers 的方法 [5]。之后，将介绍所使用的评价指标的含义，和训练得到的模型的具体表现。最后，将阐述目前模型的不足，以及可能的改进之处。

2 CNN Based Models

2.1 Overall introduction to CNNs

在 2010 年代，卷积神经网络 (Convolutional Neural Network, i.e. CNN) 一直是计算机视觉领域的 backbone。时至今日，即使基于视觉 Transformer 的模型在很多领域已经超过 CNN 成为新的 SoTA，CNN 仍然被作为很多 CV 任务的 baseline，也是入门 CV 的不二之选。

CNN 的原理很简单，总结而言，便是用卷积核来提取图片某一局部特征，再将这些卷积核一层层累在一起，最后再用几层线性层或者 1×1 的卷积层 [19] 将这些特征综合，输出为预测的结果。卷积核往往会被设计成较小的尺寸 (e.g. 3×3)，因此一个卷积核只能，其实也只需提取图片的局部特征即可。位于底层的卷积核感受野很小，即其输出的特征图所综合的原图像中的区域信息较少；而 CNN 中越往上的卷积核，深度越深，其感受野越大，综合得到的特征也更加全面和抽象。但无论如何，CNN 中的卷积层部分都是局部的特征提取器，真正综合这些特征的是之后的全连接层；这一点被之后出现的基于 Transformer 的特征提取器彻底改变。本文之后也会予以讨论。对于 CNN 更详细的解释可以参考 [6] 和 [7]。

CNN 的设计有很多非常 Tricky 的地方，于此也涌现了一大堆文章。例如在模型设计上，究竟是“宽网”好还是“深网”好，“深网”的深度究竟多深比较合适，激活函数用什么比较合适等等。以及针对训练过程中出现的问题，如何进行改进。例如如何防止过拟合，如何让模型快速收敛，如何让训练过程更加平缓等等。

2.2 Representative Works of CNNs

CNN 很早便被提出，例如 1998 的 LeNet[8] 便已经基本和目前通用的 VGG 等 CNN 的结构相同，并在手写数字识别上取得了巨大的成功。但在之后的十余年时间里，受制于当时的硬件运算速度和数据集大小的限制 (CNNs are hungry GPU and dataset consumers!)，CNN 一直未能再取得突破。直到 2012 年 AlexNet 的提出，在 ImageNet 上大幅超越了已有结果，重新点燃了 CNN 的研究热潮。AlexNet 的成功一部分归功于 GPU 硬件加速单元的出现和在机器学习领域的应用，大幅加速了模型的训练，使得更大参数量的训练变得可能；同时如 ImageNet 这样的大规模数据集的出现，也让更大的

模型不容易过拟合。而之后的机器学习模型的性能，也基本与 GPU 算力和数据集规模成正相关。

在本部分，笔者将简单介绍三个代表性的 CNN 模型：AlexNet,VGG,ResNet。相关内容参考自 [9] 以及对应论文原文。

2.2.1 AlexNet,2012 [1]

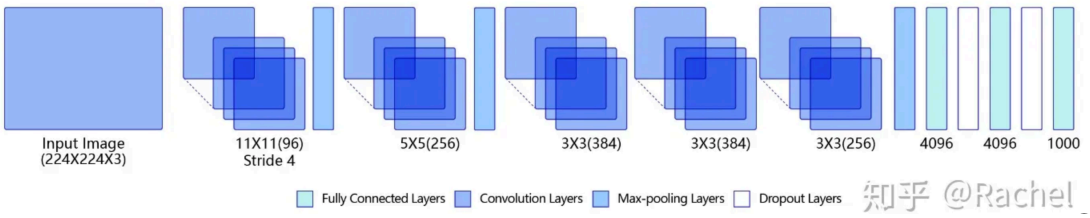

Figure 1: Architecture of AlexNet[9]

作为第一个在大规模数据集上进行图像识别取得成功的卷积神经网络，AlexNet 采取了比之前的 LeNet 更深同时更宽的网络结构：一共由五层卷积和三层全连接构成。在卷积层设计上，AlexNet 的前三层卷积层的尺寸逐渐减小，即在底层采取较大的卷积核以增大其感受野。该设计在当时被认为是有必要的，而之后 VGG 等网络则声称并证明了更大尺寸的宽网，完全可以被由更多的小尺寸的卷积核组成的深网所取代，并取得更好的结果。

在增加网络参数量之外，AlexNet 还采用了很多技巧以取得更好的效果：

- ReLU 激活函数：相较于其他复杂的激活函数，ReLU 前传和反传计算量非常小，同时也能保证梯度不会饱和；
- Dropout 防止过拟合：在全连接层之间添加 Dropout 层，即在训练过程中强行使得一定比例的神经元停止工作，以减少神经元之间的相互依赖（complex co-adaptations of neurons），迫使其学习到更加鲁棒的图像特征；
- 数据增强：在 256×256 的图像中随机选取 224×224 的图像块来进行训练。

上述三个技巧在之后被广泛应用在深度学习的网络设计中。此外，AlexNet 还引入了神经元的侧抑制（Local Response Normalization）以提高 ReLU 的表现，但之后的许多工作则证明该设计的泛化性并不好，几乎只在 AlexNet 中起作用 [2]。

2.2.2 VGG,2014 [2]

AlexNet 尽管取得了很好的效果，但并未指明 CNN 的设计方向：即究竟是深度更重要还是宽度更重要。而 2014 年的 VGG 则证明了在相同的参数量下深网可以比宽网取得更好的效果，同时模型性能几乎可以随着卷积层深度的增加线性提高。

相较于 AlexNet 使用在卷积层中使用尺寸逐渐减小的卷积核，VGG 在其卷积层中严格使用 3×3 的卷积核和池化层，并通过增加卷积核的层数而非宽度来提高模型的学习能力。

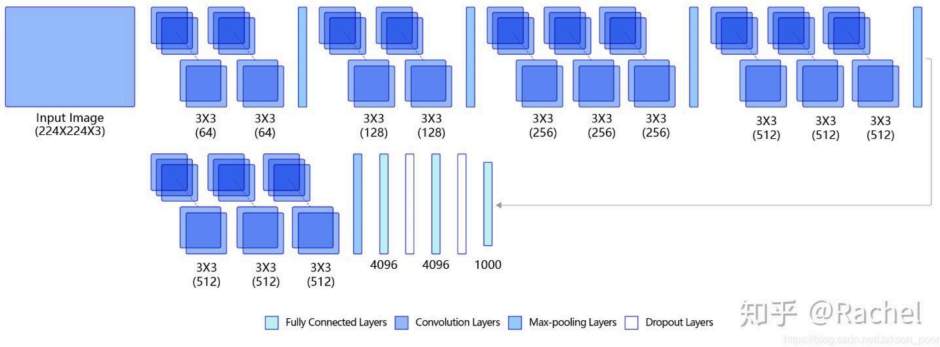

Figure 2: Architecture of VGG[9]

VGG 如此设计的重要考虑是，更多层的小卷积核能比单一层的大卷积核在保证感受野相同时，减少参数量，从而让大量增加模型的深度成为可能。例如，两层 3×3 卷积核和一层 5×5 卷积核的感受野相同，但参数只有后者的 18/25；三层 3×3 卷积核和一层 7×7 卷积核感受野相同，但参数只有后者的 27/49。

2.2.3 ResNet,2017 [3]

Depth matters, and residual networks are easier to train and optimize.

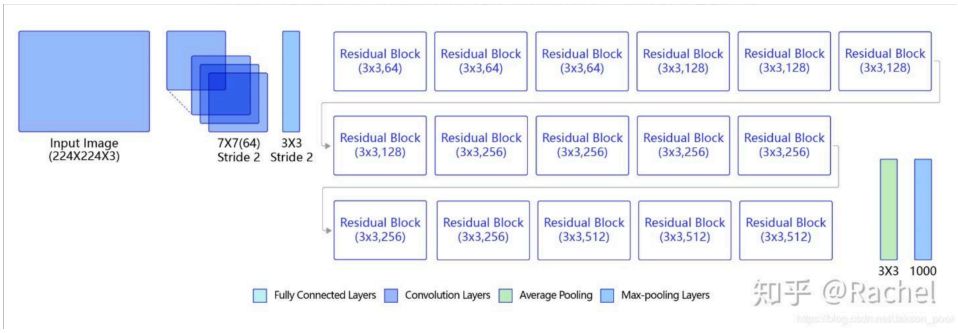

Figure 3: Architecture of ResNet[9]

VGG 的成功证明了增加网络的深度，能够有效提高模型学习特征的能力。但是之后的实践表明，随着网络深度的持续增加，模型的表现不增反降。为了解决上述问题，进一步提高深度卷积神经网络的表现，Kaiming He 等人在 2017 年提出了 ResNet。

ResNet 的核心架构为如下的残差连接块：

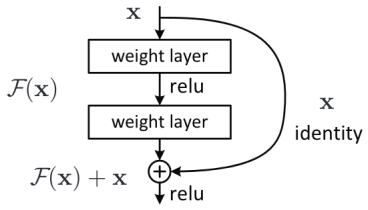

Figure 4: Residual Block [3]

相比于直接要求模型对原始数据分布进行拟合，ResNet 声称对希望结果和原始数据的差进行拟合更加容易。通过这样的残差连接块，作者轻松构建并训练出了最多达 152 层的深度学习网络，将 ImageNet 的分类错误率降低到了 3.6%，该结果甚至比一般人眼的分类准确率还要高。

3 Vision Transformers

人在用眼睛观看某一图像时，往往会倾向于关注到图像中的特定部分而非整体。例如，在看一张夜空中一轮圆月的照片时，我们第一眼便会将目光投注到月亮上去，而不是几乎全黑的背景，尽管其中可能会有些许繁星和云彩。基于 CNN 的方式只会使用训练得到的卷积核，对图像中的所有部分都执行一遍运算，尽管在上述情形中，绝大部分的卷积特征提取都对最终的图像理解毫无帮助。为了解决上述问题，一个自然的想法，便是使用注意力（Attention）机制，对原始图像中的每个部分施加一个权值，来告诉模型应当将注意力放到图片的哪一部分上去。

早在 2015 年，便有工作尝试将注意力机制引入 CV 领域。该年的 ICML 论文 [10]，将注意力机制、结合 CNN 和 RNN，引入到了 NIC(Neural Image Caption) 任务中去，让模型在每次输出 caption 的下一个词时，都会用不同的注意力权值参考原图像。例如对于”A dog is standing on a hardwood floor.” 这句标注，模型在预测 dog 这个词的时候，便将更多的注意力放到图片中的 dog 而非背景部分上。

Figure 5: More examples of attention mechanism *Show, Attend and Tell* [11]

此后的模型也同样遵循了这一范式，即仍然使用 CNN 作为特征提取器，只将网络的中间某一层替换为自注意（Self-Attention）层 [12]。或者简单将卷积核替换为所谓”Local Attention”，使用 Attention 机制来综合局部信息 [13][14]。这些模型的参数量和 CNN 基本相同，结果也仅仅稍微好于 CNN，未能打破卷积神经网络在 CV 领域的主导地位。

2017 年的神文 *Attention is All You Need*[4]，第一次使用纯 Attention 层构建的网络，在 NLP 任务上击败了一众基于 RNN 机制的 SoTA，并迅速成为自然语言模型的 backbone。随着 Transformer 在 NLP 领域的风靡，越来越多 CV 的研究人员也开始将

目光关注到 Self-Attention 机制在视觉任务的应用上。但如何很好的处理稠密的图片特征，一直困扰着研究者。

数字存储的图片和自然语言有着天壤之别：自然语言中的每一个独立的词都有着丰富的语义信息，而图片中的单一像素则几乎毫无信息量可言。只有当这些像素信息按照特定的顺序进行排列时，才能形成视觉特征。此外，一张图片往往有成万上亿的像素点，这复杂度和输入序列长度成平方关系的 Transformer 结构而言无疑是致命的。

2021 年的 ICLR 论文 [5]，使用了一个十分自然的方式来规避上述问题：将图片也视为 Words 输入到标准的 Transformer 结构中去；将原图分割为 16×16 的 Patches，处理这些小的分快而非直接处理原图像，来规避复杂度爆炸的问题。同时利用 Transformer 的 Position Embedding 机制，将 Patches 在原图中的位置信息巧妙的编码到模型中去。

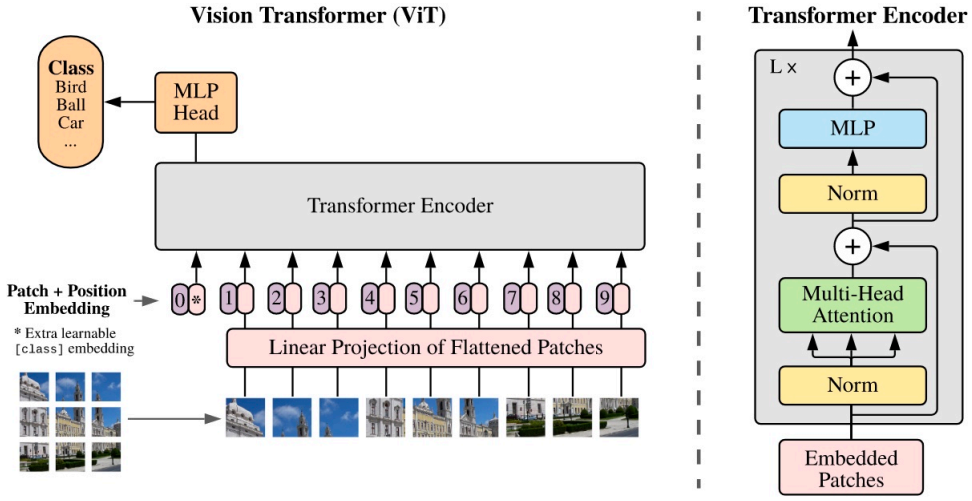

Figure 6: Architecture of ViT [5]

ViT 在小规模的数据集上并未取得很好的结果，作者认为这是由于相较于 CNN，ViT 的归纳偏置 (Inductive Bias) 更少。其相关的两段原文如下：

- In CNNs, locality, two-dimensional neighborhood structure, and translation equivariance are bakes into each layer throughout the whole model. In ViT, only MLP layers are local and translationally equivariant, while the self-attention layers are global.
- The position embeddings at initialization time carry no information about the 2D positions of the patches and all spatial relations between the patches have to be learned from scratch.

但当在大规模的数据集 (14M-300M images) 上进行预训练时，ViT 则开始超过传统的 ResNet 等 CNN 方法。同时，相较于传统的卷积操作，Transformer 所设计的基本都是经过大量优化的矩阵乘法操作，非常适合在 GPU 上进行加速，因此训练和推理过程都更快。在被提出后，ViT 开始在很多领域取代 ResNet，成为新的视觉特征提取器。

ViT 首次证明了在视觉领域单纯依靠注意力机制可以取得比卷积更好的效果。但 ViT 依然有很多缺陷：其一是模型需要在大规模的数据集上进行长时间的预训练才能

取得很好的效果，而也不是每个人都能和 Google 一样有这么多的显卡和算力；其二便是 ViT 基本只能低分辨率的图像，其将图片分割为 Patches 的方法意味着模型需要处理的序列长度，事实上仍随着图片规模的增大线性增大，导致计算复杂度依然为平方复杂度。这大大限制了 ViT 在需要 Dense Features 的领域的运用，例如实例分割、图像生成等。

针对第一点，大量针对 ViT 的训练技巧被提出 [15][16]。包括大量 Triky 的数据增强 [15]，和使用现有的 ResNet 模型进行知识蒸馏 [16]，实现了使用更少的训练成本达到甚至超越原 ViT 模型的结果。而 2021 年提出的 Swin Transformer [17] 和 2022 年的 Swin Transformer V2 [18] 则专注于解决上述的第二点问题。相较于 ViT 在整个图片上进行全局的 Self-Attention 操作，以及由此带来的平方复杂度问题，Swin Transformer 仅在非重叠的窗口内部进行注意力计算，并通过 Shifted Window 机制来沟通融合各个窗口之间的特征，从而能在不显著增加计算代价的前提下提高模型所处理图像的分辨率。

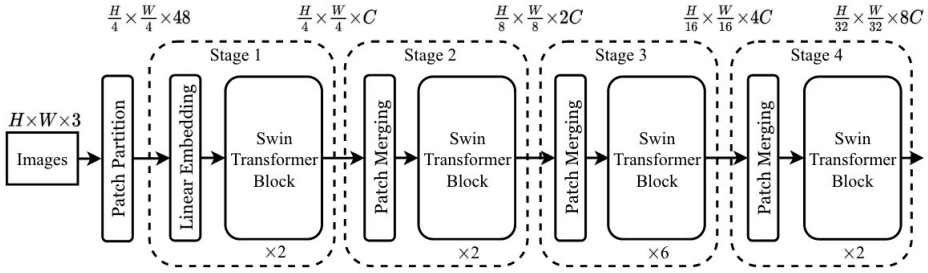

Figure 7: Architecture of Swin Transformer [18]

4 Training

本项目训练过程的相关细节如下：

- 在数据集处理部分，将原数据集按照 8:1:1 的比例划分为 Train,Val 以及独立的 Test 集；
- 在训练部分，每隔一定 step 输出当前模型的 loss 值以及在 Val 集上的准确率；
- 同时，使用 wandb 记录并绘制训练过程的 loss 值与准确率变化；
- 在每个 epoch 结束后保存模型权重，以便于后续 Evaluation。

5 Evaluation

5.1 Evaluation Indexes

对于二分类问题，采用以下评价指标：

- Accuracy（准确率）：准确率是指模型预测正确的样本数量与总样本数量的比例。它反映了模型整体分类的准确程度，包括对负样本和正样本的分类准确度。

- Precision（精确率）：精确率衡量的是模型在预测为正例的样本中，有多少是真正的正例。它表示模型在预测正例时的准确率，即模型的预测有多少是准确的。
- Recall（召回率）：召回率衡量的是模型对于真正的正例样本有多少被正确地预测为正例。它表示模型能够找出真实正例的能力，衡量了模型的查全率。
- F1 Score（F1 得分）：F1 得分综合了精确率和召回率，是精确率和召回率的调和平均值。它提供了一个综合评估分类模型性能的指标，能够同时考虑模型的准确率和查全率。
- auROC（曲线下面积）：auROC 是指 ROC 曲线下的面积，其中 ROC 曲线以真正例率（True Positive Rate，召回率）为纵轴，假正例率（False Positive Rate）为横轴。auROC 衡量了模型在不同阈值下对正例和负例的分类能力，面积越大表示模型的分类性能越好。

对于五分类与多标签分类问题，则主要使用准确率进行评价。

5.2 Results

本部分将展现最终所使用在 Test 集上进行测试的模型的表现。包括分别用于二分类和无分类的 ResNet152 和 ViT-b32 共四个模型。

5.2.1 二分类

- **评价指标对比**

二分类	ResNet152	ViT-b32
Acc	0.9238	0.9192
Precision	0.9127	0.9208
Recall	0.938	0.9186
F1 Score	0.9254	0.9197
auROC	0.9237	0.9192

- **混淆矩阵**

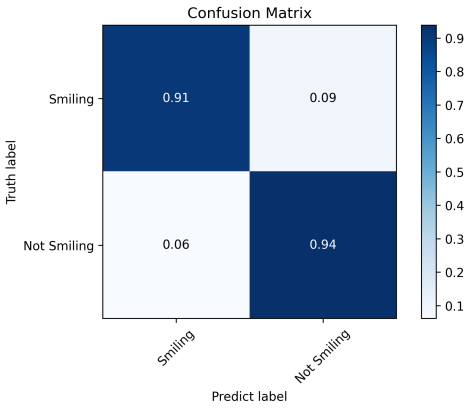

Figure 8: ResNet152

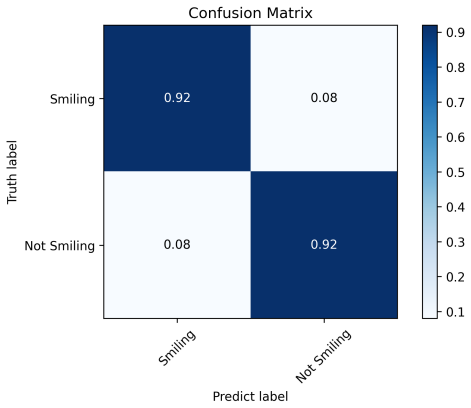

Figure 9: ViT-b32

Figure 10: Confusion Matrix

5.2.2 五分类

- **评价指标对比**

五分类	ResNet152	ViT-b32
Acc	0.8380	0.8226

- **混淆矩阵**

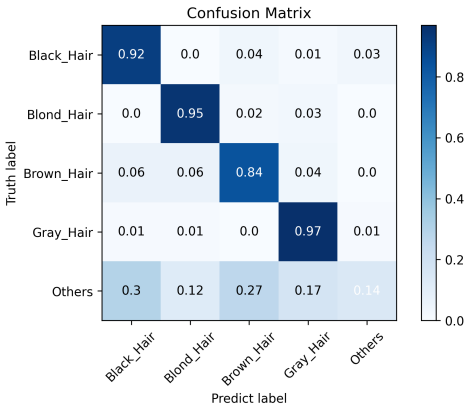

Figure 11: ResNet152

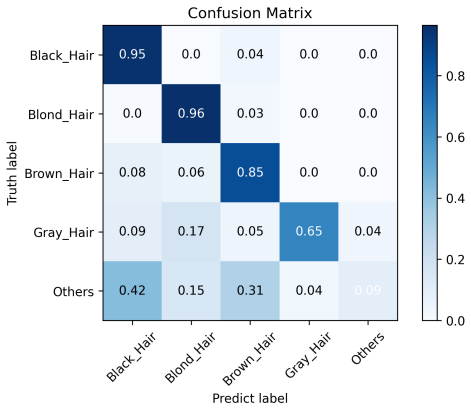

Figure 12: ViT-b32

Figure 13: Confusion Matrix

原本的数据集样本比例十分不平衡 (Black Hair: Blond Hair: Brown Hair: Gray Hair: Others=2:1:1:1:0.5)，造成

5.2.3 多标签分类

将多标签分类简单处理为对每个标签的二分类，若模型对某一标签的预测输出概率大于等于 0.5 则认为该样本具有该标签，否则则无。最后计算所有标签分类的平均准

确率为 0.9294。

5.3 Other Comparison

5.3.1 Effect of Pretraining

在二分类任务中，分别使用了非预训练和预训练的 ViT 模型在相同参数下进行训练。下面将对其进行对比：

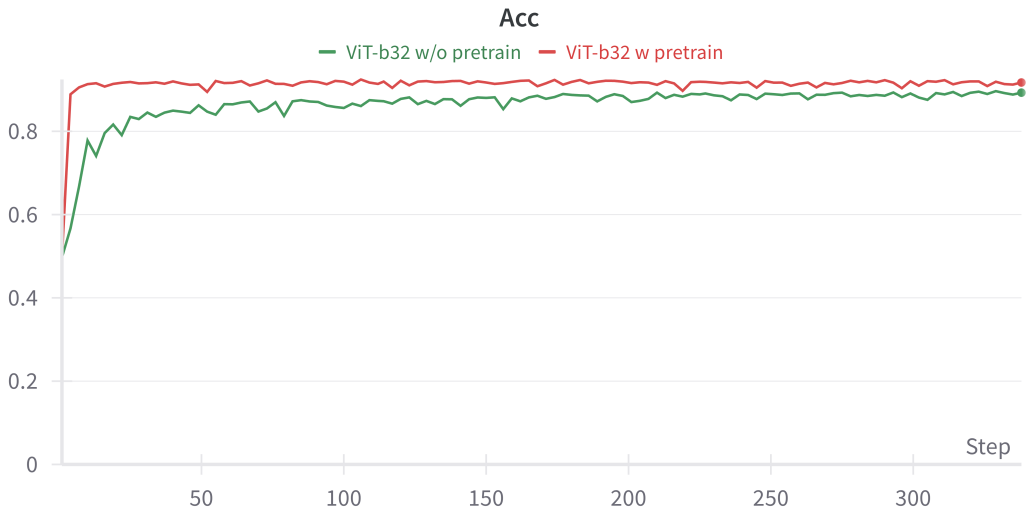

Figure 14: Acc of ViT w/o Pretraining

- 相较于非预训练模型，预训练模型的收敛速度更快；
- 大部分情况下，预训练模型最终收敛效果相较于非预训练模型更好。

5.3.2 Comparison between Different CNN Models

在二分类任务中，分别使用 NaiveCNN,AlexNet,VGG16 和 ResNet101 四种 CNN 模型。下面对其进行对比：

可以发现，CNN 模型的表现基本与网络的深度和参数量成正比。

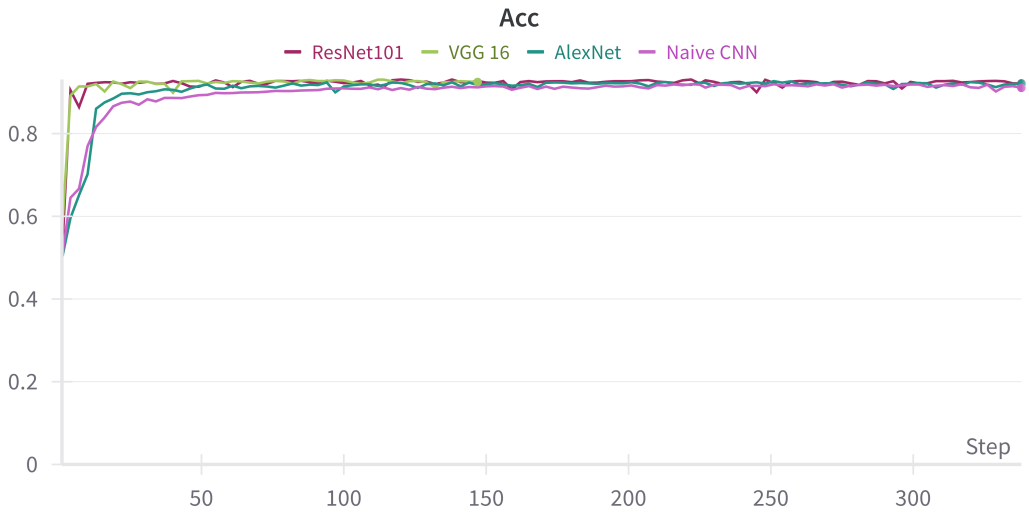

Figure 15: Accuracy of CNNs

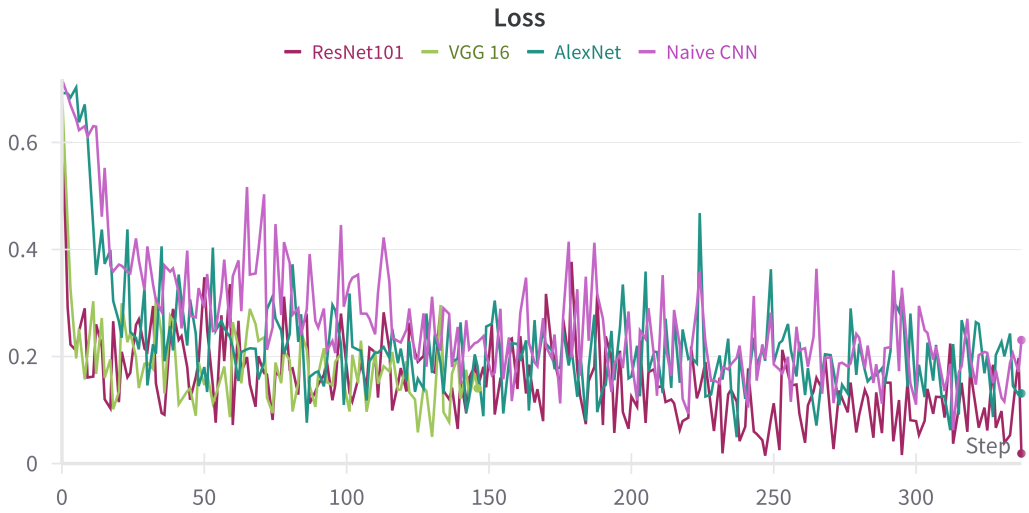

Figure 16: Loss of CNNs

6 References

References

[1] Krizhevsky, Alex, Ilya Sutskever and Geoffrey E. Hinton. "ImageNet classification with deep convolutional neural networks." Communications of the ACM 60 (2012): 84 - 90.

[2] Simonyan, Karen and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition." CoRR abs/1409.1556 (2014): n. pag.

[3] He, Kaiming, X. Zhang, Shaoqing Ren and Jian Sun. "Deep Residual Learning for Image Recognition." 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015): 770-778.

[4] Vaswani, Ashish, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser and Illia Polosukhin. "Attention is All you Need." Neural Information Processing Systems (2017).

[5] Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit and Neil Houlsby. "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale."ArXiv abs/2010.11929 (2020): n. pag.

[6] https://mp.weixin.qq.com/s/IWy5rR3vnv6-3l2sDfqYgg

[7] Zeiler, Matthew D. and Rob Fergus. "Visualizing and Understanding Convolutional Networks." ArXiv abs/1311.2901 (2013): n. pag.

[8] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learn- ing applied to document recognition. Proc. of the IEEE, 86(11):2278–2324, 1998

[9] https://zhuanlan.zhihu.com/p/162881214

[10] Xu, Ke, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, Ruslan Salakhutdinov, Richard S. Zemel and Yoshua Bengio. "Show, Attend and Tell: Neural Image Caption Generation with Visual Attention."International Conference on Machine Learning (2015).

[11] https://web.eecs.umich.edu/ justincj/teaching/eecs498/WI2022/

[12] Zhang, Han, Ian J. Goodfellow, Dimitris N. Metaxas and Augustus Odena. "Self-Attention Generative Adversarial Networks." ArXiv abs/1805.08318 (2018): n. pag.

[13] Hu, Han, Zheng Zhang, Zhenda Xie and Stephen Lin. "Local Relation Networks for Image Recognition." 2019 IEEE/CVF International Conference on Computer Vision (ICCV) (2019): 3463-3472.

[14] Ramachandran, Prajit, Niki Parmar, Ashish Vaswani, Irwan Bello, Anselm Levskaya and Jonathon Shlens. "Stand-Alone Self-Attention in Vision Models." ArXiv abs/1906.05909 (2019): n. pag.

[15] Steiner, Andreas, Alexander Kolesnikov, Xiaohua Zhai, Ross Wightman, Jakob Uszkoreit and Lucas Beyer. "How to train your ViT? Data, Augmentation, and Regularization in Vision Transformers." Trans. Mach. Learn. Res. 2022 (2021): n. pag.

[16] Hinton, Geoffrey E., Oriol Vinyals and Jeffrey Dean. "Distilling the Knowledge in a Neural Network." ArXiv abs/1503.02531 (2015): n. pag.

[17] Liu, Ze, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin and Baining Guo. "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows." 2021 IEEE/CVF International Conference on Computer Vision (ICCV) (2021): 9992-10002.

[18] Liu, Ze, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, Furu Wei and Baining Guo. "Swin Transformer V2: Scaling Up Capacity and Resolution." 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2021): 11999-12009.

[19] Shelhamer, Evan, Jonathan Long and Trevor Darrell. "Fully convolutional networks for semantic segmentation." 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2014): 3431-3440.