南開大學
Nankai University

计算机学院

深度学习大作业报告

CIFAR100 数据集分类

王子涵 2011740

王德言 1910109

杨馨仪 2011440

专业：计算机科学与技术 & 信息安全

2024 年 6 月 25 日

目录

1 项目链接 **2**

2 实验分工 **2**
2.1 论文复现 . . . 2
2.2 方法创新 . . . 2
2.3 报告撰写 . . . 2

3 论文复现 **2**
3.1 Vision Transformer(ViT) . . . 2
3.1.1 ViT 结构 . . . 2
3.1.2 ViT [1] 复现结果 . . . 3
3.2 EdgeNeXt . . . 4
3.2.1 EdgeNeXt 结构 . . . 4
3.2.2 EdgeNeXt 复现结果 . . . 5
3.3 Coordinate Attention . . . 5
3.3.1 实现步骤 . . . 5
3.4 Res2Net . . . 7
3.4.1 方法贡献 . . . 7
3.4.2 实现步骤 . . . 7

4 方法创新 **8**
4.1 学习率衰减 . . . 8
4.2 激活函数 . . . 8
4.3 池化层 . . . 9
4.4 数据增强 . . . 9
4.5 标签平滑技术 . . . 10
4.6 Res2Net 与 Coordinate Attention 结合 . . . 10

5 实验设置及结果分析 **11**
5.1 数据集介绍 . . . 11
5.2 性能评价指标 . . . 11
5.3 实验结果 . . . 12
5.4 消融实验 . . . 12
5.4.1 学习率衰减 . . . 12
5.4.2 激活函数 . . . 13
5.4.3 池化层 . . . 14
5.4.4 数据增强 . . . 15

6 总结 **16**

1 项目链接

https://gitee.com/yang-xinyi020819/deep_learning_classification_model_cifar100

2 实验分工

2.1 论文复现

- 王子涵：复现 Vision Transformer、EdgeNeXt
- 王德言：复现 Coordinate Attention
- 杨馨仪：复现 Res2Net

2.2 方法创新

- 王德言：实现数据增强，学习率的创新，尝试了不同激活函数与池化层对性能的影响
- 杨馨仪：实现标签平滑技术，Res2Net 与 Coordinate Attention 机制的结合

2.3 报告撰写

- 王子涵：负责论文复现的 Vision Transformer(ViT) 和 EdgeNeXt 部分，方法创新的学习率衰减、激活函数、池化层和数据增强部分，消融实验部分以及总结部分
- 杨馨仪：负责论文复现的 Coordinate Attention 和 Res2Net 部分，方法创新的标签平滑技术、Res2Net 与 Coordinate Attention 结合部分，以及实验的数据集介绍、性能评价指标和实验结果部分。

3 论文复现

3.1 Vision Transformer(ViT)

3.1.1 ViT 结构

ViT [1] 是 Google 团队提出的将 Transformer 应用在图像分类的模型，它的开创性工作就是将纯粹的 Transformer 模型应用到计算机视觉领域，核心思想是将图像视为一系列的“视觉单词”或“令牌”（tokens），类似于 NLP 领域的单词（words）。ViT [1] 由将图片进行 patch embedding，即将图片分成多个块（patch），然后通过 embedding，将每个 patch 展开为序列，为了进行分类任务，作者引入了一个额外的 class token，也就是输入序列的 0 位置的向量。

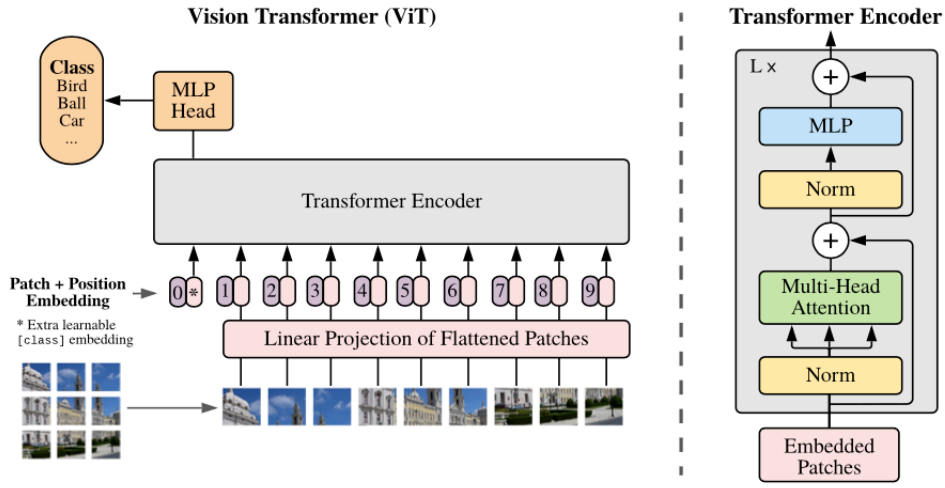


图 3.1: ViT 架构 [1]

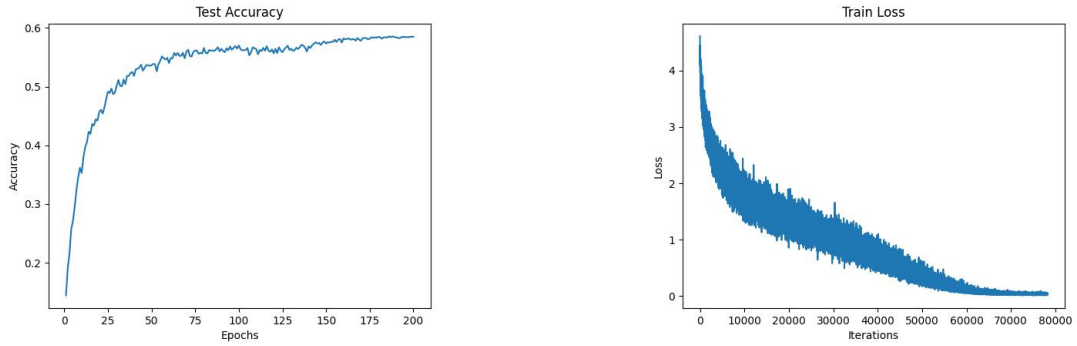
图 3.1 即为 ViT [1] 的详细架构：

- **Patch Embedding：**将图片分成 16*16 的小块（patch），然后拉长为序列进入线性投影层，线性投影层的维度 D=768*N
- **Position Embedding：**ViT 需要加入位置编码，位置编码可以理解为一张表，表一共有 N 行，N 的大小和输入序列长度相同，每一行代表一个向量。
- **Extra learnable class embedding：**特殊的 token，序号固定为 0，ViT 认为该 patch 会和所有 patch 进行两两交互从而学习到足够的知识，因此输出结果只需要输出该元素。
- **Transformer Encoder：**由 L 个 Encoder Block 堆叠而成的。Norm 层进行规范化；Attention 层是 Transformer 的核心层，这里使用多头自注意力层，即自注意力层会并行地计算多个“头”的注意力分数，计算完毕之后再进行拼接，维度最终恢复原状；MLP 层将维度放大再缩小回去（D—4D—D），引入非线性激活函数 GeLU 。

3.1.2 ViT [1] 复现结果

表 1: ViT 复现参数设置和结果

epoch	batch_size	lr	best Acc (%)
200	128	0.01	58.55



(a) ViT 复现准确率图像 (b) ViT 复现损失率图像

图 3.2: ViT 复现结果

3.2 EdgeNeXt

3.2.1 EdgeNeXt 结构

为了追求不断提高的精度，通常会开发大型复杂的神经网络，但是这种大型网络往往需要很高的计算资源，不适合部分任务使用，EdgeNeXt [6] 正是为了资源节约而设计出来的一种模型。它结合了 CNN 和 Transformer 模型的优点，构建了一个轻量级的网络结构。简单来说，它为了实现精度加入了 Transformer 模型的自注意力机制，但是自注意力通常需要复杂和长时间的计算，为了简化计算，该模型引入了分裂深度转置注意 (STDA) 编码器，该编码器将输入张量分割为多个通道组，并利用深度卷积和跨通道维度的自注意力来隐式增加感受野并编码多尺度特征。从而实现了在保持较小模型容量的同时提升模型性能的目标。

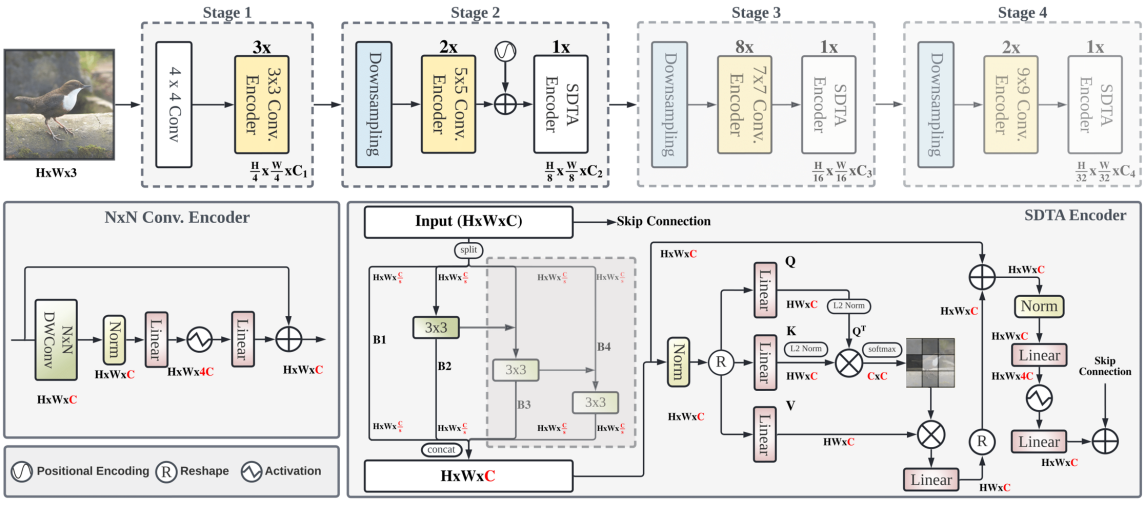


图 3.3: EdgeNeXt 架构 [6]

图 3.3 为 EdgeNeXt 核心架构：

- **第一行的四个阶段：**
  - Step1: 一张大小为 H×W×3 的输入图像经过一个初始的 patchify stem 层，该层通过一个 4×4 大小的卷积，将图像转化为更小的块（或称为 patches），形成 h/4×w/4×C1 的特征映射，然后使用三个 3×3 卷积 (Conv.) 编码器。

  - Step2-4: 第二步使用 2×2 跨步卷积，这个操作将特征映射的空间尺寸减半（H/8×W/8），同时增加通道数（C2），以便能够捕捉更丰富的信息。随后是两个连续的 5×5SDTA 编码器，它们进一步处理和增强特征。第三步和第四步同样先使用 2×2 跨步卷积，然后分别是 7×7 和 9×9SDTA 编码器。

- **conv 编码器：**通过使用 N×N 深度卷积在空间上进行特征提取和混合，随后利用两个点卷积在通道间进行特征整合和混合，从而实现了对输入数据的高效特征提取和变换。
- **SDTA 编码器：**SDTA 编码器将输入的张量分割为多个通道组，并利用深度卷积网络和跨通道的自注意力来隐式地增加感受野，并编码多尺度特征。这种设计在提高模型性能的同时，减少了计算复杂度。

3.2.2 EdgeNeXt 复现结果

表 2: EdgeNeXt 复现参数设置和结果

model(epoch=300)	Top-1 Acc(%)	Top-5 Acc(%)	Best Acc (%)
EdgeNeXt-small(batch_size=256;lr=0.01)	62.740	83.650	63.06
EdgeNeXt-base(batch_size=32;lr=0.001)	64.950	84.480	65.05

根据实验结果：我尝试的一些参数调整对精度提升效果有限，但是基本都优于 ViT [1] 的训练成果，ViT [1] 更适用于数据量比较大的模型，训练数据越多效果越好，而 CIFAR100 [5] 是个比较小的数据集。相对于 ViT [1]，EdgeNeXt [6] 模型基于 ViT [1] 做了轻量化处理，更适合小数据集。

3.3 Coordinate Attention

Coordinate Attention [4] 是一种新颖的注意力机制，旨在将位置信息嵌入到通道注意力中。与传统的通道注意力通过 2D 全局池化将特征张量转换为单个特征向量不同，Coordinate Attention [4] 将通道注意力分解为两个 1D 特征编码过程，分别沿水平方向和垂直方向聚合特征。这样可以在一个空间方向上捕获长距离依赖关系，同时在另一个空间方向上保留精确的位置信息。最终生成的方向感知和位置敏感的注意力图可以互补地应用于输入特征图，从而增强对感兴趣对象的表示。该机制简单易用，且几乎不增加计算开销，能够灵活地嵌入到经典的移动网络结构中。

3.3.1 实现步骤

1. **特征聚合**：

   首先，给定输入特征图 X，采用两个 1D 全局池化操作分别沿水平和垂直方向聚合每个通道的特征。具体来说，对于第 c 个通道在高度 h 处的输出 $z_c^h(h)$ 计算如下：

$$z_c^h(h) = \frac{1}{W} \sum_{0 \le i < W} x_c(h, i).$$

   同样地，第 c 个通道在宽度 w 处的输出 $z_c^w(w)$ 计算如下：

$$z_c^w(w) = \frac{1}{H} \sum_{0 \le j < H} x_c(j, w).$$

这两个变换分别沿两个空间方向聚合特征，产生一对方向感知的特征图。

2. **特征融合与压缩**：

将这两个方向感知的特征图拼接在一起，然后通过一个共享的 1×1 卷积变换函数 F_1 进行处理，得到中间特征图 f：

$$\mathbf{f} = \delta\left(F_1\left(\left[\mathbf{z}^h, \mathbf{z}^w\right]\right)\right)$$

其中 $[\cdot, \cdot]$ 表示沿空间维度的拼接操作，δ 是一个非线性激活函数，f 的尺寸为 $\mathbb{R}^{C/r\times(H+W)}$，$r$ 是控制块大小的缩减率。

3. **特征分割与再变换**：

将 f 沿空间维度分割为两个独立的张量 f^h 和 f^w，并分别通过两个 1×1 卷积变换函数 F_h 和 F_w 转换为与输入 X 具有相同通道数的张量 g^h 和 g^w：

$$\mathbf{g}^h = \sigma\left(F_h\left(\mathbf{f}^h\right)\right),$$
$$\mathbf{g}^w = \sigma\left(F_w\left(\mathbf{f}^w\right)\right).$$

其中，σ 是 sigmoid 函数。

4. **注意力权重应用**：

最终，将 g^h 和 g^w 分别扩展并用作注意力权重，输出 Y 可以表示为：

$$y_c(i,j) = x_c(i,j) \times g_c^h(i) \times g_c^w(j) \tag{1}$$

这样通过结合方向感知和位置信息生成的注意力图，可以更准确地捕捉视觉任务中的长距离依赖关系和物体的空间结构。

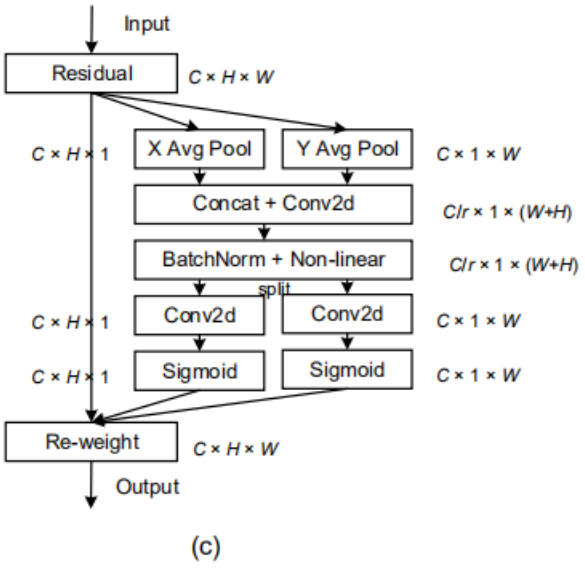


图 3.4: Coordinate Attention 模块示意图 [4]

3.4 Res2Net

在众多视觉任务中，多尺度特征表示具有极其重要的意义。近年来，卷积神经网络的进展不断展示出更强的多尺度表示能力，从而在各种应用中取得了持续的性能提升。然而，大多数现有方法是在层级上表示多尺度特征。Res2Net [2] 是一种用于 CNN 的新型构建模块，它通过在单个残差块内构建分层类残差连接来表示多尺度特征。Res2Net 在更细粒度的层次上表示多尺度特征，并增加了每一网络层的感受野范围。此外 Res2Net [2] 模块可以嵌入到当前最先进的主干 CNN 模型中，例如 ResNet [3]、ResNeXt [8] 等等。

3.4.1 方法贡献

- **多尺度特征表示：** Res2Net [2] 将输入特征图分成多个等大小的组，并在每个组上进行独立的卷积操作，然后将这些组的卷积结果逐步融合。这种设计使其输出包含了不同感受野大小的组合，允许每个残差块内部捕获到多尺度特征，增强了对全局和本地信息的提取能力。
- **特征重用和参数减少：** Res2Net [2] 模块通过省略第一个分割部分 X_1 的卷积，将其直接传递到后续层，从而实现了特征重用。这不仅减少了计算量，还降低了参数数量，提高了模型的效率和性能。
- **引入新的尺度控制参数：** Res2Net [2] 引入了一个新的尺度控制参数 s，即模块中特征组的数量。通过调整 s 值，可以灵活控制模型的多尺度表示能力，适应不同的任务需求。例如，在处理高分辨率图像或复杂场景时，可以选择较大的 s 值，以捕获更多的细节信息；而在处理低分辨率或简单场景时，可以选择较小的 s 值，以减少计算量。
- **高效的计算和内存开销：** 相比于其他增加层数或宽度来提升多尺度特征表示能力的方法，Res2Net [2] 通过在单个残差块内实现多尺度处理，极大地减少了额外的计算和内存需求。这种高效性使得 Res2Net [2] 非常适合在资源有限的环境中应用，如移动设备和嵌入式系统。

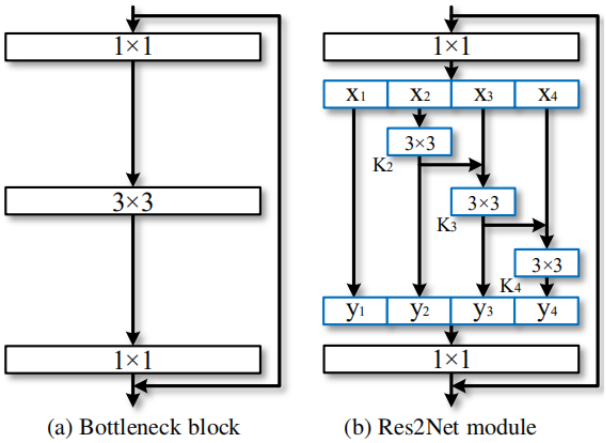


图 3.5: res2net-基本块 [2]

3.4.2 实现步骤

1. **特征图分割**：

- 将经过 1×1 卷积的特征图均匀地分割成 s 个子特征图集合 $\{x_i\}_{i=1}^{s}$，每个子集 x_i 具有相同的空间尺寸，但通道数是输入特征图的 $\frac{1}{s}$。

2. **特征融合**：

   除了第一个子特征图 x_1 外的每个子特征图 x_i（其中 $i > 1$），都有一个对应的 3×3 卷积操作 K_i。这个操作 K_i 会产生一个输出 y_i。

   - 对于除了子特征图 x_1 和 x_2 外的每个 x_i，将其与上一个输出 y_{i-1} 相加，然后通过 K_i 进行处理。
   - 对于 x_2，直接通过 K_i 进行处理。
   - 对于 x_1，由于其通常拥有更高的维度（即保留了更多的通道信息），因此不需要再经过额外的 3×3 卷积，直接作为 y_1 输出。

$$\mathbf{y}_i = \begin{cases} \mathbf{x}_i & i = 1 \\ \mathbf{K}_i\left(\mathbf{x}_i\right) & i = 2 \\ \mathbf{K}_i\left(\mathbf{x}_i + \mathbf{y}_{i-1}\right) & 2 < i \leqslant s \end{cases}$$

3. **特征拼接与降维**：

   - 最后，将所有经过处理的特征图 $\{y_i\}_{i=1}^{s}$ 拼接在一起，并通过一个 1×1 卷积操作，将通道数降低到与输入特征图相同的维度。

4 方法创新

4.1 学习率衰减

学习率（Learning Rate）是神经网络训练中一个非常重要的超参数，它决定了在每一次迭代中参数更新的幅度。在梯度下降算法中，学习率决定了参数沿着梯度反方向移动的步长。如果学习率设置得过大，可能会导致训练过程不稳定，模型在最优解附近震荡而无法收敛；如果学习率设置得过小，虽然能保证模型收敛，但可能会导致收敛速度过慢，需要更多的迭代次数。

学习率衰减有助于模型在训练初期快速找到大致的方向，然后在训练后期精细调整参数，以达到更好的性能。在训练初期，模型对参数空间的搜索范围较大，使用较大的学习率可以使模型更快地找到大致的最优解区域，随着训练的进行，模型逐渐接近最优解，此时再使用较大的学习率可能会导致模型在最优解附近震荡而无法稳定收敛，通过逐渐减小学习率，可以使模型在最优解附近进行更精细的调整，从而得到更好的性能，并且在训练后期，随着学习率的减小，模型对训练数据的拟合能力逐渐减弱，从而减少了过拟合的风险。

因此通过学习率衰减可以加速收敛、避免震荡和提高泛化能力。

基于该改进思路，我们采用了每五个 epoch 衰减 1/2 的学习率衰减算法，实验结果请参考第五部分的消融实验。

4.2 激活函数

激活函数的主要作用之一是加入非线性因素，以解决线性模型表达能力不足的缺陷。另一个重要作用是执行数据的归一化，将输入数据映射到某个范围内，再往下传递，这样可以限制数据的扩张，防

止数据过大导致的溢出风险。

对于深度学习模型来说，激活函数是不可或缺的一环，常用的激活函数有十多种，不同的激活函数具有不同的特性和优点，能够适应不同的任务和数据集，例如，tanh 函数常用于自然语言处理和语音识别任务的递归神经网络中，而 ReLU 函数则在深度学习中被广泛使用，并展现出优异的性能和通用性。通过调整激活函数，可以找到最适合当前任务的函数，从而提升模型的性能。

基于此改进思路，我们分别尝试了除 Sigmoid 以外多种激活函数，包括 Relu,elu,Softmax 等，实验结果请参考第五部分的消融实验。

4.3 池化层

池化层通常位于卷积层之后，主要作用是对输入的特征图进行下采样，即减小特征图的尺寸，同时保留重要的特征信息。

池化层主要有两种类型：最大池化和平均池化。最大池化在特征图的每个池化窗口中选择最大值作为输出，有助于保留输入特征图中的显著特征；平均池化在特征图的每个池化窗口中计算所有值的平均值作为输出，有助于保留特征图的全局信息，并减少噪声的影响。

使用不同的池化层会产生不同的训练效果，我们尝试使用不同类型的池化层，以找到最适合当前任务的池化策略，尝试的池化层类型：

- **Avg+Max：** 结合了平均池化和最大池化两种方式，同时获取特征的全局信息和显著特征，从而提供更丰富的特征表示。
- **Avg+L1：** 结合平均池化并在池化窗口内计算 L1 范数作为输出。L1 范数对异常值或显著特征较为敏感，在某些情况下可以强调特征图中的稀疏性。
- **Avg+L2：** 结合平均池化并在池化窗口内计算 L2 范数作为输出。L2 池化可以增强模型的鲁棒性，因为它对特征图中的异常值或噪声更为稳健。

详细实验结果参考第五部分的消融实验。

4.4 数据增强

数据增强数据增强也叫数据扩增，意思是在不实质性的增加数据的情况下，让有限的数据产生等价于更多数据的价值。在实际训练中，训练数据不足或数据分布不均的问题可能导致模型在训练过程中出现过拟合现象，数据增强通过生成更多的训练样本，使得模型能够接触到更多的数据模式，从而减少过拟合的风险。

数据增强的核心思想是通过增加训练样本的多样性和数量，帮助模型更好地学习到数据的本质特征，比较常见的数据增强手段包括几何变换、颜色变换、添加噪声等。

基于此改进思路，我们采取了以下数据增强手段，轮流在数据增强处理后的训练集和原训练集上训练以减少过拟合。代码如下：

Listing 1: 数据增强

```
transform_dataEnhance = transforms.Compose([
transforms.RandomHorizontalFlip(),  # 随机水平翻转
transforms.RandomVerticalFlip(),  # 随机水平翻转
transforms.ColorJitter(brightness=0.2, contrast=0.2, saturation=0.2, hue=0.2),  # 调整颜色
```

```
    transforms.ToTensor(),                    # 转换为Tensor
    transforms.Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225])  # 标准化
])
```

4.5 标签平滑技术

标签平滑技术 [7] 是常用于分类任务中的正则化方法，它可以避免模型在训练过程中过于相信自己的预测结果，而导致的训练效果的下降。具体来说，在一般的模型训练的过程中，期待模型的预测标签向量应当为独热编码，预测结果应当是“非黑即白”的。然而，有时一些样本可能具备一些其他类别的特征，这些特征可能会反映到其他类别的概率上去。如果完全忽略这些可能性，则会导致模型的泛化性能下降，当面临未见过的新数据时，不能准确地预测出正确的类别。

标签平滑技术 [7] 的解决方法非常直观，就是将预测的标签从硬标签平滑为软标签。软标签是硬标签与均匀分布加权平均得到的。采用这样的方法，训练过程中就不会过于相信样本属于某个类别，而考虑到预测结果中属于其他类别的可能性，这样的操作降低了发生过拟合的可能性。标签平滑技术 [7] 的公式如下所示：

$$\hat{y}_i = \mathrm{y}_{\mathrm{hot}}(1-\alpha) + \alpha/K$$

如图4.6所示给出了一些常见的标签向量形式，可供参考：

Categories		A	B	C	D	E
Hard Label	One-hot	1	0	0	0	0
	Multi-hot	1	1	0	1	0
Soft Label	Label Smoothing	0.9	0.025	0.025	0.025	0.025
	Ideal Distribution	0.45	0.1	0.05	0.35	0.05

图 4.6: 相关标签形式展示

4.6 Res2Net 与 Coordinate Attention 结合

向 Res2Net 结构中引入 Coordinate Attention 中能够有效弥补 Res2Net 在保留精确位置信息和建模复杂通道依赖关系方面的不足。Coordinate Attention 通过将位置信息嵌入通道注意力中，利用两个独立的 1D 特征编码过程分别沿垂直和水平方向聚合特征，既能够捕捉长距离的空间依赖关系，又能保持位置信息的精确性，从而提升模型在视觉任务中的性能和泛化能力，同时保持较低的计算复杂度。

具体来说，Coordinate Attention 模块加在了每一个残差块的最后一个 1×1 卷积层和 BatchNorm 层之后、残差连接之前。当 att_type 为”CA” 时，通过 CoordAtt 类实现该模块的加入，对特征图进行加权处理。这一步骤增强了模型对空间位置信息和通道关系的感知能力，进一步提升了网络在视觉任务中的性能表现。

5 实验设置及结果分析

5.1 数据集介绍

CIFAR-100 [5] 数据集是一个由 60,000 张 32x32 彩色图像组成的计算机视觉数据集，其中包括 50,000 张训练图像和 10,000 张测试图像，涵盖 100 个细分类别和 20 个超类别，每张图像都有细分类别标签和超类别标签。它常用于图像分类和机器学习研究，因其较低的分辨率和高类别相似度，具有较高的挑战性。CIFAR-100 [5] 数据集可以通过多种深度学习框架方便地加载和使用，广泛应用于模型训练、迁移学习和算法评估等领域。

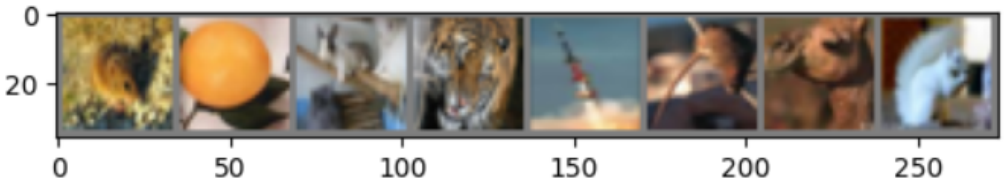


图 5.7: CIFAR100 [5] 数据集部分图像展示

5.2 性能评价指标

在本实验中，我们使用了多个评价指标来衡量模型的性能，包括 Top-1 Accuracy、Top-5 Accuracy 和 Best Accuracy。这些指标能够全面反映模型在图像分类任务中的表现。

1. **Top-1 Accuracy**

   Top-1 准确率是指模型预测的类别与真实类别完全匹配的准确率。具体来说，模型会为每个输入图像输出一个概率分布，Top-1 准确率计算的是模型预测概率最高的类别是否与真实类别相同。

2. **Top-5 Accuracy**

   Top-5 准确率是指模型预测的前五个最可能的类别中是否包含真实类别。在这种情况下，只要真实类别出现在模型预测的前五个最高概率的类别中，就认为该预测是正确的。Top-5 准确率特别适用于像 CIFAR100 [5] 一样有较多类别的分类任务，能够提供更宽容的评价标准，更好地反映模型在实际应用中的表现。

3. **Best Accuracy**

   最佳准确率通常是指模型在验证集或测试集上达到的最高准确率。具体来说，可以是指训练过程中在某一时刻达到的最佳 Top-1 准确率。它能够帮助我们选择和保存性能最优的模型进行部署，确保在实际应用中使用的模型具有最佳的表现。

5.3 实验结果

表 3: 实验结果

model(epoch=50)	Top-1 Acc(%)	Top-5 Acc(%)	Best Acc (%)
ResNet [3]	49.300	78.540	49.750
Res2Net [2]	50.760	79.520	51.650
ResNeXt [8]	50.930	79.560	51.620
ViT [1]	/	/	53.620
EdgeNeXt [6]	54.460	80.030	53.840
ResNetCA	60.740	86.860	62.530
Res2NetCA_v1b(Ours)	**64.660**	**88.430**	**66.590**

综合分析实验结果显示,我们提出的 Res2Net 与 Coordinate Attention 相结合的模型 Res2NetCA_v1b 在图像分类任务中取得了显著的性能提升。其中 v1b 代表在模型开始增添了两层卷积层，在每个残差块内增加了一层平均池化层。

具体而言，该模型达到了 64.66% 的 Top-1 准确率和 88.43% 的 Top-5 准确率，而最佳准确率为 66.59%。相比之下，传统的 ResNet、Res2Net 和 ResNeXt 模型的表现略逊一筹，而新兴的 EdgeNeXt 和 ViT 模型虽然表现出相对较高的性能水平，但由于其模型更为复杂，需要更多的计算资源来实现其优势，这在一些情况下限制了它们的实际应用，导致他们的准确率也低于 Res2NetCA_v1b。

另外，通过对比 ResNet 和 ResNetCA 可以发现，Coordinate Attention 模块对于位置信息的提取在图像分类任务上展现出较为优秀的表现，ResNetCA 在 Top-1 准确率上超出了 ResNet 11.44%。对比 Res2Net 和 ResNet 也可以证明引入尺度控制参数后，模型在提取多尺度特征信息上展现出优势，性能超越了 ResNet。

我们的模型 Res2NetCA_v1b 结合前文提到的多种创新方法，超越了实现的全部基线方法，验证了方法的有效性，证实了其在多尺度特征融合和准确的位置信息处理方面具有优势，适合应对复杂的图像分类任务。

5.4 消融实验

本部分的消融实验设置的模型和参数为——baseline：CoordAtt；depth=18；epoch=15；lr=0.1。原模型的训练结果：Top-1 accuracy 53.86%；Top-5 accuracy 81.1%。

下面将分别对学习率衰减、激活函数、池化层和数据增强四个修改方向进行消融实验结果分析。

5.4.1 学习率衰减

采用每五个 epoch 减半的指数衰减学习率，实验结果如下：

(a) 学习率衰减Top1 (b) 学习率衰减Top5

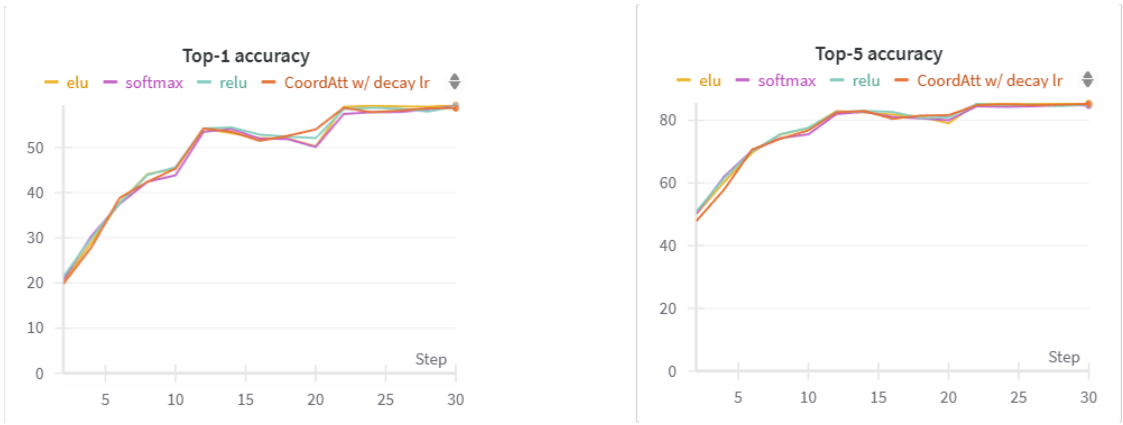
图 5.8: 学习率衰减结果

采用学习率衰减算法之后，Top-1 accuracy 达到 58.75%，Top-5 accuracy 达到 85.25%，相较于原模型 Top-1 的精度提高了 4.89 个百分点，Top-5 的精度提高了 1.15 个百分点。

Top-1 准确率的显著提升表明学习率衰减算法有助于模型更好地拟合训练数据，并提高了对测试数据的泛化能力，说明此改进的高效性，虽然 Top-5 准确率的提升相对于 Top-1 准确率来说较小，但这也是一个积极的信号。Top-5 准确率的提升表明模型在预测时不仅能够更准确地识别出最可能的类别，还能够更准确地识别出前五个最可能的类别。这可能是因为学习率衰减使得模型在训练过程中能够更好地学习到数据之间的内在关系，从而提高了对多个类别的预测能力。

5.4.2 激活函数

在添加学习率衰减的基础上尝试使用不同的激活函数，分别为 Sigmoid、ReLU、ELU 和 Softmax，实验结果如下：



(a) 激活函数-Top1 (b) 激活函数-Top5

图 5.9: 激活函数对比结果

表 4: 激活函数结果表格

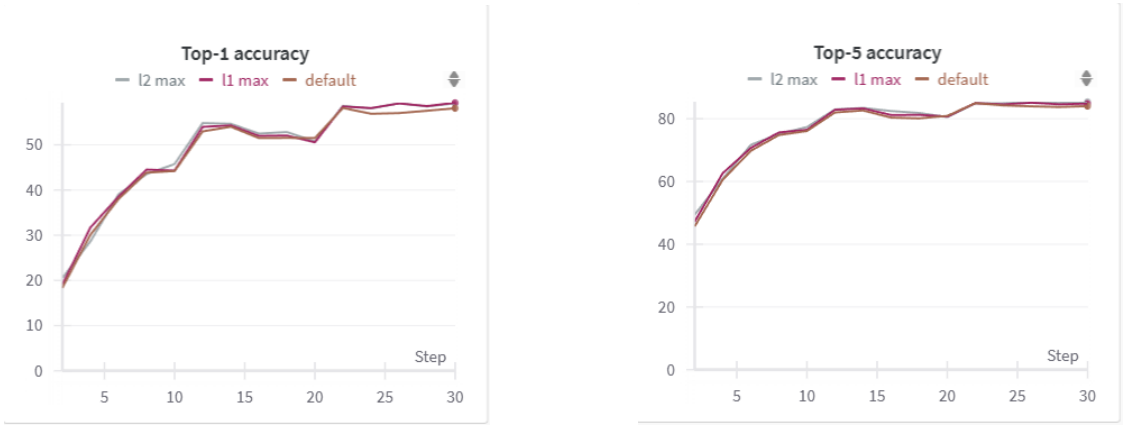
激活函数	Top-1 Acc(%)	Top-5 Acc(%)
Sigmoid	58.75	85.25
ReLU	59.08	85.11
ELU	59.45	85.56
Softmax	59.19	84.77

根据实验结果可以看出，这几种激活函数的性能相差的并不多。从数据方面来看，原模型使用的是 Sigmoid 函数，相较于原模型，ReLU 函数和 Softmax 函数在 Top-1 的精度上优于 Sigmoid，但是在 Top-5 的精度上都劣于 Sigmoid。原因可能是在 Top-1 精度上，ReLU 和 Softmax 函数能够准确地预测最可能的类别，Softmax 通过强调最可能的类别来优化 Top-1 精度，所以 Softmax 在 Top-1 上的精度优于 ReLU 和 Sigmoid。而 Sigmoid 函数允许更平滑的概率分布，这使得模型在预测 Top-5 类别时更加灵活和准确。

只有 ELU 函数在 Top-1 和 Top-5 上的训练结果都比 sigmoid 优秀，分别增加了 0.7 和 0.31 个百分点，说明对于这个任务和数据集，ELU 函数具有更好的非线性拟合能力、梯度传播特性和计算效率，能够更准确地学习到数据的分布和特征。

5.4.3 池化层

修改通道池化层的池化方式，基于 avg+max 的池化层，尝试 avg+L1 和 avg+L2 两种池化层，实验结果如下：



(a) 池化层-Top1 (b) 池化层-Top5

图 5.10: 池化层修改结果

表 5: 池化层结果表格

池化层类型	Top-1 Acc(%)	Top-5 Acc(%)
baseline(AVG+MAX)	58.09	84.02
AVG+L1	59.32	84.81
AVG+L2	59.18	85.41

我们尝试的两种新型池化层相较于 baseline 对性能都有所提升。AVG+L1 的 Top-1 准确度提升

了 1.23 个百分点，Top-5 的准确度提升了 0.79 个百分点；AVG+L2 的 Top-1 准确度提升了 1.09 个百分点，Top-5 的准确度提升了 1.39 个百分点；

从 Top-1 准确率来看，AVG+L1 池化方式略微优于 AVG+MAX 和 AVG+L2，这说明 AVG+L1 在捕捉最显著特征方面更有效，从而提高了模型对最可能类别的预测能力；从 Top-5 准确率来看，AVG+L2 池化方式表现最好，这可能意味着 AVG+L2 在捕捉更广泛的信息方面更有效，从而提高了模型对前五个最可能类别的预测能力。

这两种池化层各有优劣，并没有拥有绝对的性能优势。

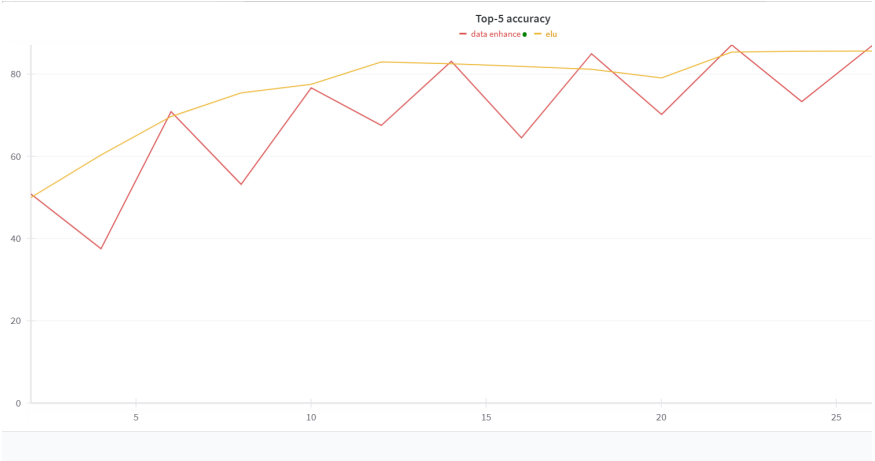
5.4.4 数据增强

在学习率衰减以及选择 ELU 作为激活函数的基础上进行数据增强处理，处理操作参考 4.4 部分的数据增强代码，然后轮流使用原始数据和增强数据进行训练以减少过拟合，增加模型泛化能力。

数据增强后，Top-1 accuracy 提升到 60.42%，相较于 baseline 提升了 1.67 个百分点，Top-5 accuracy 提升到 86.88%，相较于 baseline 提升了 1.63 个百分点，因此我们认为数据增强确实对模型性能有所改进

训练结果如下：

(a) 数据增强-Top1



(b) 数据增强-Top5

图 5.11: 数据增强结果

6 总结

我们小组本次深度学习大作业的主要任务是模型复现和模型改进，模型复现选取 Vision Transformer（ViT）模型、EdgeNeXt 模型、Coordinate Attention 模型以及 Res2net 模型进行复现，模型改进则选取 Res2net 模型作为主干模型，针对性地提出改进策略并尝试将 Coordinate Attention 机制和 Res2net 模型进行融合。

在模型复现选取的几个模型中，ResNetCA 复现效果是最好的，Top-1 Acc 达到 60.740%，Top-5 Acc 达到 86.860%，该模型是基于 Resnet 的修改版，在模型基础上加入了 Coordinate Attention，提高了原模型对于位置信息的提取。

在模型改进中，我们提出了学习率衰减、优化激活函数、修改池化层、进行数据增强、标签平滑技术处理以及将 Coordinate Attention 机制融入 Res2net 模型等创新点，并分别做了消融实验，获得了良好的改进效果，作为 baseline 的 Res2net 的 Top-1 Acc 为 50.76%，Top-5 Acc 为 79.52%，Best Acc 为 51.65%，在经过一系列改进过后，Top-1 Acc 为 64.66%，提升了 13.9 个百分点，Top-5 Acc 为 88.43%，提升了 8.91 个百分点，Best Acc 为 66.59%，提升了 14.94 个百分点，性能提升效果非常明显，相比较于复现效果最好的 ResnetCA 也有很大的提升，说明我们的改进策略非常成功，其中将 Coordinate Attention 机制融入 Res2net 模型的策略获得的效果最好，也说明了自注意力机制在图像分类上确实拥有很高的优越性。

在本次大作业的实践过程中，我们研读了许多计算机视觉图像处理领域的经典论文，学习了许多经典模型以及模型每一部分的功能，并通过改进模型更深入地掌握了这些知识，能够从整体和从局部观察研究模型，并且培养了提出 idea 的能力，这是一次收获满满的旅程!

参考文献

[1] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[2] Shang-Hua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip Torr. Res2net: A new multi-scale backbone architecture. *IEEE transactions on pattern analysis and machine intelligence*, 43(2):652–662, 2019.

[3] 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.

[4] Qibin Hou, Daquan Zhou, and Jiashi Feng. Coordinate attention for efficient mobile network design. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13713–13722, 2021.

[5] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[6] Muhammad Maaz, Abdelrahman Shaker, Hisham Cholakkal, Salman Khan, Syed Waqas Zamir, Rao Muhammad Anwer, and Fahad Shahbaz Khan. Edgenext: efficiently amalgamated cnn-transformer architecture for mobile vision applications. In *European Conference on Computer Vision*, pages 3–20. Springer, 2022.

[7] Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. When does label smoothing help? *Advances in neural information processing systems*, 32, 2019.

[8] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017.