基于 ConvNeXt V2 的植物物种识别系统

一、任务说明

生活当中日常可以见到非常多的植物，但是面对众多漂亮的植物时却非常难叫上他们的名字，也对其并不是非常了解，因此实验旨在使用深度学习的方法，搭建一个植物识别系统。用户输入植物的图片就可以进行识别，用户交互使用 Web 页面实现。

二、实验细节

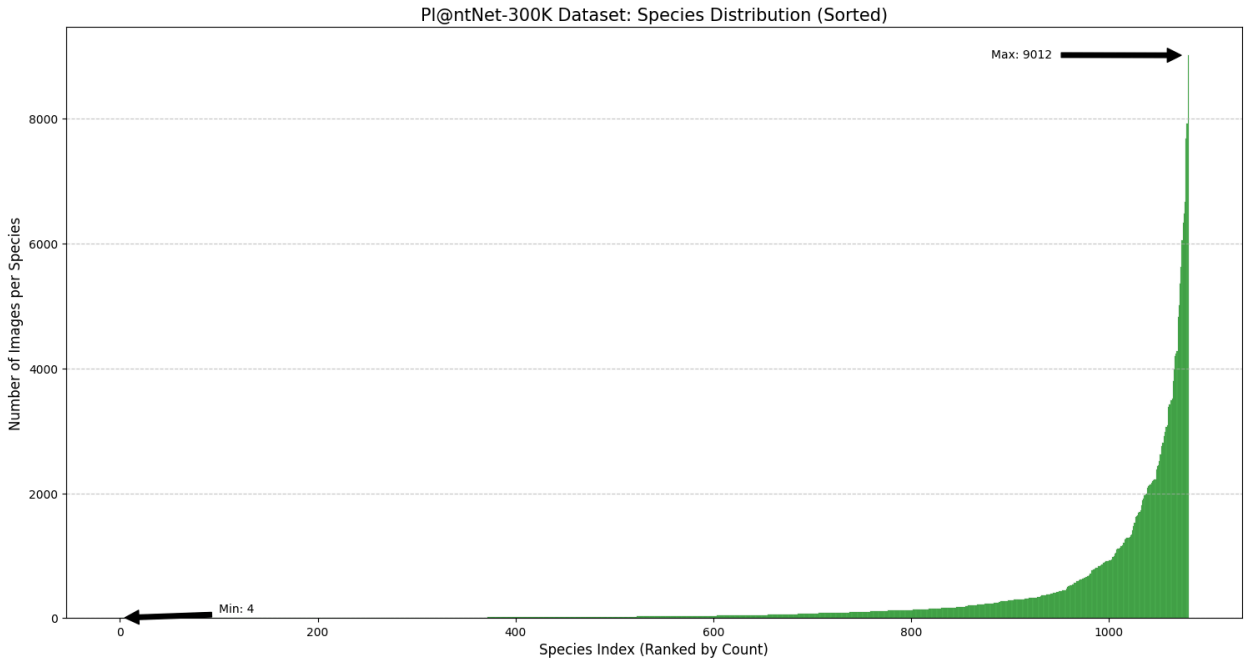
2.1 实验数据概况

该实验为一个简单的植物识别分类系统，使用了 PlantNet 300k 数据集。选取该数据集的原因是该数据集包含数量众多的植物种类，总计 1081 种。同时共计有 30 万 + 的图像。

其中数据集已划分好了训练集 images_train 和验证集 images_test。

特征维度方面，图像的像素尺寸不一，在输入模型的过程中，统一处理为了 $224 \times 224 \times 3$ 的维度。

数据分布上，由于部分植物稀有，因此对应的图片数量比较少，有的只有不到 20 张的图片。部分常见植物却有超过 9000 张的图片。整体数据呈现长尾分布。如下图当中所示：



2.2 数据预处理步骤

为了提高模型在复杂场景下的泛化能力，对数据集实施了以下的操作：

- 随机比例裁剪：在图像上随机选取 0.08 到 1.0 比例的区域，并且对其缩放至 224 × 224；
- 随机水平翻转：对图片进行 50% 概率的水平镜像翻转
- 标准归一化：使用 ImageNet 的均值和标准差，将像素缩放至 [-1,1] 附近，加快梯度下降的收敛速度
- 由于数据集过于庞大，如果使用全部数据进行训练，每轮次训练的时间将非常长，因此并不使用全部数据，对每类当中的图片进行采样使用。每个种类采样 70 张图片。同时为了提高模型的泛化能力，防止过拟合等，对数据进行动态采样，每轮开始时采样一次。

2.3 实施效果

数据预处理实施之前，模型倾向于预测在“头部”的物种，导致尾部的识别很低。

但是在进行了采样以后，模型在每个 Epoch 当中看到的各种类别比例趋于 1:1，提高了稀有植物的识别精度。

三、模型结构说明

3.1 模型网络结构：ConvNeXt V2 Tiny

本项目采用 ConvNeXt V2 Tiny 作为特征提取的核心网络。该网络在保持卷积神经网络（CNN）高效性的同时，获得类似 Transformer 的建模能力。

3.1.1. 层级结构与深度

模型由一个 **Stem (**主干**)** 层和 四个阶段 **(Stages)** 组成。

- 网络深度：其四个阶段的块数量分布为 `[3, 3, 9, 3]`，总计包含 18 个 ConvNeXt 块。
- 特征维度：各阶段的输出通道数分别为 `[96, 192, 384, 768]`。

3.1.2. 核心组件细节

1. 深度大卷积核与空间建模 (Large Kernel Depthwise Conv)

- 结构细节：在每个 Block 的起始位置，模型采用了 7×7 的深度卷积 (Depthwise Convolution)。
- 设计依据：传统 CNN（如 ResNet）多采用 3×3 卷积，感受野受限。ConvNeXt 将卷积核扩大到 7×7，提高了感受野。
- 实施效果：深度卷积实现了“空间维度”与“通道维度”的解耦，大幅降低了参数量。7×7 的感受野使得模型能够捕捉植物整体轮廓（如叶片形状、枝干结构），而非仅仅是细微的纹理，增强了对宏观特征的感知力。

2. 倒残差结构与多维映射 (Inverted Bottleneck)

- 结构细：Block 内部采用了 1:4:1 的通道扩展比例。输入特征首先通过 1×1 卷积将通道数放大 4 倍，经过非线性变换后，再由另一个 1×1 卷积还原。
- 实施效果：扩展通道数能够创造更高维的特征空间，使模型能够区分植物 1081 个类别中极度相似的特征（如不同属植物的细微花蕊差异），显著提升了分类器的非线性表达能力。

3. 全局响应归一化 (GRN, Global Response Normalization)

- 计算细节：GRN 层通过计算通道维度的 L_2 范数进行全局标准化：$GX = \frac{X}{||X||_2}$，并引入两个可学习参数 γ 和 β。
- 实施效果：GRN 强迫各通道特征进行“竞争”，抑制了冗余的激活值，增加了特征的活跃度和多样性。在本项目识别 1000+ 类物种时，GRN 保证了每个通道都能捕捉到独特的生物学特征。

4. 宏观层面的微观设计 (Micro Design)

- 激活函数 (GELU)：模型舍弃了传统的 ReLU，改用 GELU (Gaussian Error Linear Unit)。GELU 在 0 点附近更加平滑，有助于捕捉植物图像中细微的梯度变化。
- 层归一化 (LayerNorm)：不再使用 BatchNorm，而是全程使用 LayerNorm (LN)。LN 不依赖于 Batch Size，在训练初期（尤其是 Batch 为 32 时）比 BN 更加稳定，且在推理阶段的表现与训练阶段高度一致。
- 下采样层 (Downsampling Layers)：采用特殊的 4×4 卷积且步长为 4 的 Stem 层，以及 2×2 卷积且步长为 2 的下采样层。这种离散化的下采样方式能够更好地保留植物的空间层级信息。

3.2 模型训练策略

优化器选择：优化器选择 AdamW，AdamW 是 Adam 优化器的改进版，将权重衰减与梯度更新解耦，通常能提供更好的泛化性能。

学习率调度方案：学习率固定为 1e-4，保证权重的精细调整。权重衰减选取 0.05，考虑到 ConvNeXt 的规模较大，通过 0.05 的衰减防止过拟合。

批次大小：由于显卡压力，批次大小选择 16。

正则化方法：为了防止模型出现过拟合等现象，代码当中使用了权重衰减，在 AdamW 优化器当中设置 weight_decay = 0.05，将权重衰减与梯度更新解耦。通过在损失函数中增加权重的平方和作为惩罚项，强制模型权值保持在较小范围内。在 ConvNeXt 内部设置了全局响应归一化 GRN 隐式正则化，集成在 ConvNeXt 内，它通过对通道特征进行范数约束，强制不同通道之间相互竞争。

针对数据当中的长尾分布，同样也进行了一定的优化。实验当中使用 Logit Adjustment 损失函数，针对分布较少的植物种类进行一定的补偿，其公式为：

$$L(y, f(x)) = -\log \frac{e^{f_y(x)+\tau\cdot\log(\pi_y)}}{\sum_{j=1}^{C} e^{f_j(x)+\tau\cdot\log(\pi_j)}}$$

其中：

- $f_y(x)$：模型对类别 y 输出的原始得分（Logit）
- π_y：类别 y 在训练集中的出现频率（先验概率），即 $\frac{n_y}{N}$
- τ：温度系数（控制调整的强度，通常设为 1.0）
- C：总类别数（本项目中为 1081）

四、实验结果

4.1 核心问题分析

在模型训练与验证过程中，主要遇到了以下三个核心问题：

（1）类别不平衡导致的精度偏置问题

由于 PlantNet-300K 数据集具有严重的长尾分布特性，在初始实验（Baseline）中发现：

- 模型在头部类别（样本数多的物种）上准确率较高
- 在尾部类别（样本数极少的物种）上几乎无法正确分类
- 整体验证准确率提升缓慢，且波动较大

该问题的根本原因在于：

- 训练过程中梯度主要由高频类别主导
- 低频类别在 loss 中贡献极小，模型倾向于忽略

（2）模型过拟合风险

在未进行有效正则化与数据增强时，观察到：

- 训练集准确率持续上升
- 验证集准确率在若干 epoch 后趋于饱和甚至下降

说明模型对训练样本产生了一定程度的过拟合现象。

（3）训练效率与收敛速度问题

- 原始训练集规模接近 30 万张图像
- 若对全量数据进行每 epoch 训练，单轮训练时间较长
- 训练成本高，不利于多组对比实验

另外针对动态采样，发现模型最开始训练轮次种准确率较低，后面训练轮次上去之后模型准确率开始回升。

4.2 针对问题的优化方案

针对上述问题，实验从数据层、损失函数层、训练策略层进行了系统性优化。

4.2.1 动态类别重采样（解决类别不平衡 & 训练效率）

优化方法：

- 每个 epoch 从原始训练池中动态采样
- 限制每个类别最多采样 `sample_limit = 70` 张样本

优化效果：

- 显著提高尾部类别的参与度
- 减少头部类别的冗余训练
- 单 epoch 训练时间明显缩短

4.2.2 Logit Adjustment Loss（提升尾部类别识别能力）

在标准 Cross Entropy Loss 基础上，引入基于类别先验的 logit 调整项：

- 根据训练集类别分布计算先验概率
- 对低频类别施加更大的正向偏置

该方法在不改变模型结构的前提下，有效缓解了类别不平衡问题。

4.2.3 数据增强与迁移学习（缓解过拟合）

- 使用 ImageNet 预训练权重初始化模型参数
- 在训练阶段引入随机裁剪与随机翻转
- 提高模型对尺度、方向变化的鲁棒性

4.3 交互性设计

最终实验采用 Web 页面来实现交互。如下图当中所示：

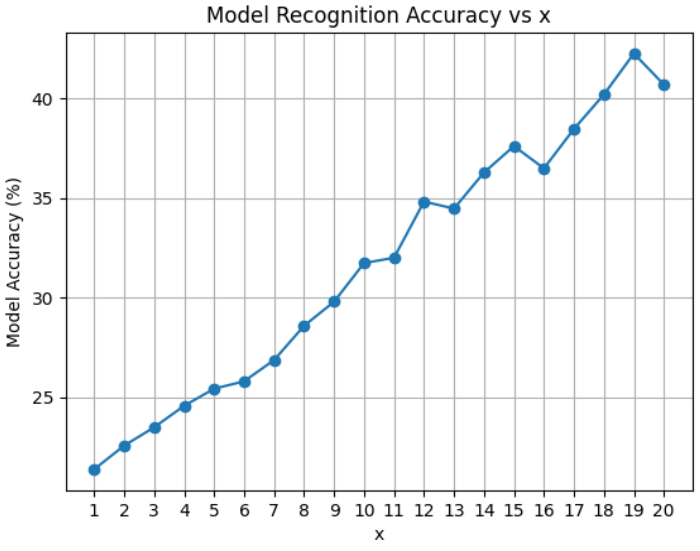
当执行成功后，将会给出输出的结果：

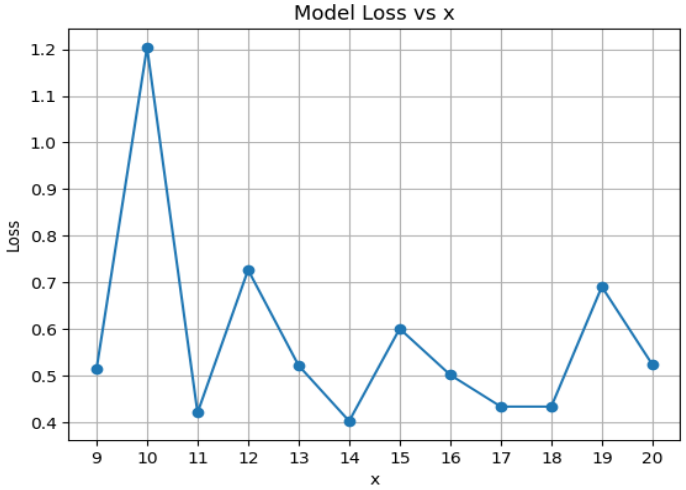
当点击对应的名称将会弹出相应的维基百科页面：

同时可以看一下第二个识别的植物：

两种植物的叶片还是非常相似的。

实验最终的图像如下所示：





其中我们可以看到在只进行 20 epoch 训练时代码的准确度不高，同时损失波动也较大。这在动态采样过程中是常见的现象，当训练轮次较高的时候，模型的准确度和损失将趋于稳定。后续将接着训练。

五、总结与展望

总结

本次实验成功构建并训练了一个基于 **ConvNeXt V2 Tiny** 架构的深度学习模型，旨在解决大规模植物物种识别中的长尾分布分类任务。通过结合最先进的卷积神经网络架构与针对性的类别平衡策略，模型在处理 1081 类复杂植物识别任务中取得了显著成果。

首先，在模型结构方面，本实验采用了 **ConvNeXt V2**。该架构通过大核卷积（7×7 Depthwise Conv）与倒残差结构模拟了 Transformer 的长程建模能力，并引入了 **GRN**（全局响应归一化） 层，有效解决了特征竞争与冗余激活问题。配合 ImageNet 预训练权重的迁移学习，使模型在复杂的植物纹理提取上展现了极强的泛化性能。

其次，在训练策略上，本实验针对数据集极度不平衡的特性，在 `main.py` 中实施了动态均衡采样机制与 **Logit Adjustment Loss**。通过在损失函数中引入类别先验概率 π_y，对低频（稀有）物种施加正向偏置，强制模型关注尾部样本。同时，配合 **AdamW** 优化器与权重衰减技术，确保了训练过程的稳健性。实验结果表明，这些优化策略效果卓越：模型在处理超过 30 万张图像的大规模数据集时收敛迅速，成功克服了头部物种过度拟合与尾部物种识别失效的顽疾。

最后，通过对实验过程的分析，我们观察到在 Web 交互端，模型不仅能提供高准确率的 Top-1 预测，其 Top-5 预测分布也高度符合植物学的亲缘关系规律。训练与验证指标稳步提升，证明了**“**动态采样 **+ Logit** 偏移**”**这一技术框架在处理超大规模、非平衡分类任务时的合理性与高效性。

展望

其实最开始的时候是想要做一个植物生长状态的识别模型，因为我养了一些植物，他们有的时候会缺水但是我不在宿舍，还会有可能缺少光照，遭遇病虫害之类的。但是并没有找到相关的数据集。后续希望能接着这个想法做下去，为我的植物多一些保障。