广东东软学院

学生实验报告

**实验课程名称：《机器学习》**

**实验项目名称：计算机视觉之花朵分类**

**实验类型：综合性**

**指导教师：杨雨倩**

**实验日期：2024 年 12 月 23日**

|  |  |  |  |
| --- | --- | --- | --- |
| **学生姓名** | **孔令军** | **学 号** | **22216920213** |
| **班 级** | **2班** | **专业名称** | **计算机科学与技术** |
| **实验组**  **其他成员** |  | | |
| **实验地点** | **B312** | | |
| **实验成绩**  **（教师签名）** |  | | |

|  |
| --- |
| **实验目的与要求**  **目标**  目标：使卷积神经网络能对5种图片进行分类   1. **项目简介（5分）**   本项目旨在构建一个基于卷积神经网络（CNN）的花朵图像分类模型。通过使用TensorFlow和Keras深度学习框架，我们从Google服务器下载了一个包含五种不同种类花朵的照片数据集，并将其分为训练集和验证集。然后，我们创建并优化了数据处理管道，以确保高效的数据读取和预处理。接着，我们设计了一个包含多个卷积层、池化层以及全连接层的CNN架构，用于提取花朵图片的特征并进行分类。在模型编译阶段，选择了Adam优化算法和稀疏分类交叉熵作为损失函数，同时将准确率设定为评估指标。经过10轮次的训练后，我们得到了模型在训练集和验证集上的性能表现，并通过绘制准确率和损失的变化图表来分析模型的学习过程。   1. **数据集介绍****（5分）**   数据集是从Google的服务器下载的一个花朵图片集合，具体来源于TensorFlow提供的示例图像资源。这个数据集是通过以下URL获取的它包含了五种不同类型的花朵照片，每一种类型都有各自的文件夹存放其相关的图像文件。  在代码中，我们首先使用函数来下载并解压这个.tgz格式的数据包到本地存储。下载完成后，我们通过对象指定了数据目录，并且确认了该数据集中包含的所有JPEG格式图像的数量，以此验证数据是否正确加载。 |
| 1.下载数据集（5分）  通过指定的URL从Google Cloud Storage下载了一个压缩文件（.tgz格式），该文件包含了一组分类好的花朵照片。下载过程由TensorFlow的tf.keras.utils.get\_file函数处理，这个函数会检查本地缓存中是否已经存在相同文件，如果不存在，则从给定的URL下载文件，并根据指示自动解压。下载完成后，程序将解压后的文件夹路径转换为pathlib.Path对象，这是一种面向对象的方式处理文件路径，可以更方便地进行各种路径操作。接着，通过移除文件扩展名，确保得到的是解压后实际存放图像的目录路径。    2.查看数据集中的图片信息，展示数据集中的图片（5分）        3.使用 Keras中的image\_dataset\_from\_directory从磁盘加载数据，并进行训练集80%和验证集20%划分。（10分）    通过调用image\_dataset\_from\_directory函数两次，一次用于创建训练数据集（train\_ds），另一次用于创建验证数据集（val\_ds）。这两个数据集都是从同一个基础目录data\_dir中生成的，该目录包含了所有用于训练模型的花朵图像。  4.学会数据可视化及应用，展示训练数据集中的前九张数据。（5分）      5.对数据进行标准化处理。（5分）      6.基于Keras创建模型。（20分）          **1.定义模型**：构建了一个Sequential模型，包括图像归一化层、三层卷积与池化层组合（用以提取特征）、一个展平层（将多维特征图转换为一维向量）、一个全连接层以及输出层。模型最后一层的神经元数量等于类别数，适用于多类分类任务。  **2.编译模型**：配置了Adam优化器、SparseCategoricalCrossentropy损失函数（适用于多类分类且接受未经变换的logits值），并选择了准确率作为评估指标。  **3.查看模型概要**：通过调用model.summary()打印出模型结构和参数信息，帮助开发者了解模型的复杂度。  **4.训练模型**：设置了10个训练周期（epochs），使用指定的训练数据集train\_ds进行模型训练，并在每个epoch结束时利用验证数据集val\_ds评估模型性能。训练历史被保存在history对象中，可用于后续分析模型的学习曲线和表现。  7.训练模型，并显示可视化训练结果。（10分）    8.如何进行数据增强，来增加数据集？（10分）  **数据增强**  首先，代码创建了一个名为data\_augmentation的数据增强管道，它包含以下操作：  **随机水平翻转**：以50%的概率对输入图像进行水平翻转。这一操作增加了数据集的多样性，有助于模型学习到更多不变性特征。  **随机旋转**：在[-10%, +10%]的范围内随机旋转图像。这模拟了不同角度拍摄的情况，增强了模型对角度变化的鲁棒性。  **随机缩放**：在[-10%, +10%]的范围内随机缩放图像。此操作帮助模型适应目标大小的变化。  这些数据增强操作被封装在一个Sequential模型中，可以直接集成到主CNN模型中，确保每次训练时都自动应用这些变换。  **改进的CNN模型定义**  接下来，代码定义了一个改进的CNN模型，该模型包括以下层次结构：  **数据增强层**：作为模型的第一层，确保所有输入图像都会经过数据增强处理，从而提供更丰富的训练样本。  **归一化层**：将图像像素值从[0, 255]缩放到[0, 1]之间，为后续的卷积层提供合适范围的输入。  **三层卷积与池化层组合**：每层卷积层后跟随一个最大池化层，逐步提取和压缩图像特征，同时减少空间维度。  **Dropout层**：在展平层之前加入了一个Dropout层，随机丢弃20%的神经元，有效防止过拟合，提高模型泛化能力。  **全连接层**：一个具有128个神经元的全连接层继续处理和组合来自卷积层的特征信息。  **输出层**：最后一层的神经元数量等于类别数（num\_classes），输出原始的logits值，这些值可以通过softmax等激活函数转换为概率分布，用于多类分类任务。          9.增加数据集之后的可视化训练结果（10分） |
| 预测结果，要求训练和测试的准确率都不能低于60%。要求**使用训练好的模型对未知类别图像，**进行预测。**（10分）** |
| **3.收获与体会**  **数据增强的重要性**  **增加数据多样性**：通过引入随机水平翻转、旋转和缩放等数据增强技术，我深刻认识到这些操作可以显著增加训练数据的多样性。这不仅有助于防止模型过拟合，还提高了模型对不同拍摄条件（如角度、大小变化）的鲁棒性。  **提升泛化能力**：数据增强使得模型能够在训练过程中接触到更多样的样本，从而增强了它在未见过的数据上的表现。实际应用中，这种改进对于提高模型的泛化能力和可靠性至关重要。  **模型结构优化**  **Dropout层的作用**：在模型中加入Dropout层是一个重要的优化措施。它通过随机丢弃一部分神经元，有效地减少了模型的复杂度，降低了过拟合的风险。实验结果显示，Dropout层确实有助于提升模型的稳定性和预测准确性。  **多层卷积与池化的组合**：通过堆叠多个卷积层和最大池化层，模型能够逐步提取出更加抽象和复杂的特征。每一层都能捕捉到不同的模式和细节，而池化层则帮助压缩信息并减少计算量。这种设计极大地提升了模型对图像特征的理解深度。  **编译配置与评估指标**  **选择合适的优化器和损失函数**：Adam优化器以其自适应学习率的特点成为我的首选，它在加速训练过程的同时保证了良好的收敛性。而对于多类分类任务，SparseCategoricalCrossentropy作为损失函数非常适合，并且设置from\_logits=True确保了输出logits值可以直接用于后续的激活函数。  **准确率作为主要评估指标**：将准确率设为评估模型性能的主要指标，直观地反映了模型分类的正确比例。然而，我也意识到在某些情况下，仅依赖准确率可能不足以全面评估模型的表现，未来需要考虑引入更多元化的评估标准。 |
| **4.实验总结**  本次实验旨在构建一个高效的卷积神经网络（CNN）模型，用于花朵图像分类任务。通过引入数据增强技术和优化模型结构，我们希望提升模型的泛化能力和抗过拟合能力，从而在实际应用中获得更好的性能表现。数据来源：使用了TensorFlow提供的flower\_photos.tgz数据集，该数据集包含了五种不同类型的花朵图片。数据增强：为了增加数据多样性并提高模型鲁棒性，我们在训练过程中引入了随机水平翻转、旋转和缩放等数据增强技术。这些操作被封装在一个Sequential模型中，确保每次训练时都自动应用这些变换。归一化处理：将所有输入图像像素值从[0, 255]缩放到[0, 1]之间，以适应后续卷积层的要求。模型架构：构建了一个包含三层卷积与池化层组合的CNN模型，并在展平层之前加入了一个Dropout层，以防止过拟合。最后一层为全连接层，输出类别数等于数据集中花朵种类的数量。  数据增强集成：将数据增强层直接作为模型的第一层，使得整个模型能够在训练过程中自动生成多样化的训练样本。优化器：选择了Adam优化算法，它能够加速训练过程并找到更优解。损失函数：使用了SparseCategoricalCrossentropy，适用于多类分类问题，并设置了from\_logits=True以接受未经激活的logits值。  评估指标：主要关注准确率（accuracy），同时记录了训练和验证过程中的损失变化情况。训练设置：设置了10个训练周期（epochs），并在每个epoch结束时利用验证数据集评估模型性能。可视化效果：通过绘制数据增强后的图像以及训练过程中的损失和准确率曲线，直观地展示了数据增强的效果及模型的学习进度。  性能提升：引入数据增强技术和Dropout层后，模型在验证集上的表现显著改善，准确率明显提高，且表现出更强的泛化能力。数据增强的重要性：通过实践证明，数据增强不仅增加了训练数据的多样性，还提高了模型对不同拍摄条件的鲁棒性和泛化能力。模型结构优化：合理设计模型结构，如堆叠多个卷积层和最大池化层，并加入Dropout层，可以有效减少过拟合并提升模型性能。  编译配置与评估指标选择：正确选择优化器、损失函数和评估指标对于训练过程至关重要，它们直接影响到模型的收敛速度和最终性能。  实践中的挑战与解决方案：在调试过程中遇到了一些挑战，如超参数调整和内存管理等问题。通过不断尝试和查阅资料，逐步找到了有效的解决方案，如合理设置批量大小、利用缓存和预取技术优化数据管道等。  本次实验成功实现了预期目标，构建了一个高效且具有良好泛化能力的花朵分类模型。通过引入数据增强技术和优化模型结构，显著提升了模型的性能。未来的工作可以进一步探索更多元化的评估标准，尝试不同的模型架构，并考虑将模型部署到实际应用场景中，以检验其真实环境下的表现。 |