

**2024年《机器学习》**

**工程报告**

**（个人版）**



**课 程：** 机器学习

**姓 名：** 周作为

**学 号：** 2022217590

**完成时间：** 2024.5.19

|  |  |  |  |
| --- | --- | --- | --- |
| **“机器学习-大作业”评分细则** | | | |
| **成绩等级** | **具体表现** | **教师评分** | |
| 优秀（100-90] | 报告撰写优秀，题目本身难度大，工作量饱满；Notebook代码清晰准确，实验与可视化优秀 | □ |  |
| 良好（90-80] | 报告撰写良好，题目本身难度较大，工作量较多；Notebook代码较好，实验与可视化良好 | □ |  |
| 中等（80-70] | 报告撰写中等，题目本身难度一般，工作量达标；Notebook代码一般，实验与可视化达标 | □ |  |
| 及格（70-60] | 能完成基本要求，工作量较少，非Notebook代码 | □ |  |
| 不及格（<60） | 未达最低要求或抄袭线上线下资源 | □ |  |

教师签名：

一． 自己所作工作的简介

**1. 分工**

**数据准备与预处理**：

负责收集和整理数据集，包括EOSINOPHIL、LYMPHOCYTE、MONOCYTE、NEUTROPHIL的样本。进行数据清洗和预处理，如图像缩放、归一化等。

**模型训练与评估**：

负责选择合适的机器学习或深度学习模型进行训练。编写代码实现模型的训练和预测。使用交叉验证等技术优化模型参数。

**结果展示与分析**：

负责将模型评估结果以图表或报告的形式展示。分析模型的性能，提出改进建议。

**2. 设计**

**模型设计**：

选择或设计合适的机器学习或深度学习模型，如卷积神经网络（CNN）用于图像分类。设计模型的架构，包括层数、每层的神经元数量、激活函数等。

**评估设计**：

设计评估指标，如准确率、精度、召回率、F1分数等。设计实验流程，包括数据加载、模型训练、预测和评估等步骤。

**3. 使用方法**

**数据加载与预处理**：

使用数据加载器（如ImageDataGenerator）从磁盘读取数据并进行预处理。将数据划分为训练集、验证集和测试集。

**模型评估**：

加载测试集数据，使用训练好的模型进行预测。计算评估指标，如准确率、分类报告等。

**结果展示**：

使用matplotlib和seaborn绘制混淆矩阵热力图。打印分类报告和准确率等评估指标。

二、研究背景与意义

**研究背景与意义**

在医学和生物信息学领域，细胞分类是一项至关重要的任务。细胞分类的准确性对于疾病的诊断、治疗方案的制定以及药物研发等具有深远影响。随着图像处理和机器学习技术的飞速发展，自动化细胞分类系统成为研究的热点。本项目旨在利用这些先进技术，实现对细胞图像中EOSINOPHIL（嗜酸性粒细胞）、LYMPHOCYTE（淋巴细胞）、MONOCYTE（单核细胞）和NEUTROPHIL（中性粒细胞）等细胞类型的自动分类。

**主要涉及领域**

本项目涉及的主要领域包括医学图像处理、机器学习和深度学习。医学图像处理技术用于从细胞图像中提取有用的特征；机器学习和深度学习技术则用于构建分类模型，实现对细胞类型的自动识别和分类。

**主要研究方法**

1. **数据预处理**：对细胞图像进行必要的预处理，如缩放、裁剪、灰度化、去噪等，以提高图像质量和模型性能。
2. **特征提取**：利用传统的图像处理技术或深度学习模型（如卷积神经网络CNN）从细胞图像中提取有用的特征。这些特征可以是形状、纹理、颜色、边缘等低层次特征，也可以是深度学习模型学习到的深层次特征。
3. **模型构建**：基于提取的特征，构建分类模型。可以采用传统的机器学习算法（如支持向量机SVM、随机森林等），也可以采用深度学习模型（如CNN、ResNet、EfficientNet等）。
4. **模型训练与优化**：使用训练集数据对模型进行训练，并使用验证集数据进行模型调优。可以采用交叉验证等技术来评估模型的性能，并根据评估结果调整模型参数。
5. **模型评估**：使用测试集数据对训练好的模型进行评估，计算准确率、精度、召回率、F1分数等指标，以评估模型的性能。

**主要存在问题**

1. **数据不平衡**：不同细胞类型的样本数量可能存在较大差异，导致模型在训练过程中对某些类别的样本过拟合，而对其他类别的样本欠拟合。
2. **特征提取困难**：细胞图像中的特征可能因光照、噪声、细胞重叠等因素而变得复杂且难以提取。
3. **模型泛化能力**：模型在训练集上表现良好，但在测试集上性能下降，泛化能力不足。

三、模型方法

这个Sequential模型定义了一个卷积神经网络，用于处理图像分类任务。下面是对该模型中用到的各个方法和理论的详细说明：

**1. 卷积层**

* **作用**：用于提取图像中的特征。
* **参数**：
  + filters：卷积核的数量，决定了输出特征图的深度。
  + kernel\_size：卷积核的大小，定义了感受野的大小。
  + strides：卷积核在图像上滑动的步长。
  + activation：激活函数，用于增加非线性。
  + padding：是否使用零填充来保持输出尺寸。

**2. 批量归一化层**

* **作用**：对神经网络的每一层输出进行标准化处理，使得每一层的输出具有相近的分布。这有助于加速训练过程、提高模型性能，并减少对初始化权重的敏感性。

**3. 最大池化层**

* **作用**：对特征图进行下采样，减少空间维度（高度和宽度），同时保留最重要的特征。这有助于减少计算量，并增加感受野的大小。
* **参数**：pool\_size定义了池化操作的窗口大小。

**4. Flatten层**

* **作用**：将多维的输入一维化，常用于从卷积层到全连接层的过渡。

**5. 全连接层**

* **作用**：在Flatten层之后，全连接层用于整合前面提取的特征，并输出最终的分类结果。
* **参数**：
  + units：输出神经元的数量。
  + activation：激活函数，对于输出层，如果是多分类问题，通常使用softmax；对于其他层，可以使用relu等激活函数。

**6. Dropout层**

* **作用**：在训练过程中，随机将网络中的一部分神经元“丢弃”掉，以减少过拟合。
* **参数**：rate定义了丢弃神经元的比例。

**7. 激活函数**

* **ReLU**：relu激活函数，对于所有输入值小于0的，输出为0；对于输入值大于0的，输出等于输入值。它有助于解决梯度消失问题，并加速训练。
* **Softmax**：softmax激活函数通常用于多分类问题的输出层，它将神经元的输出转换为概率分布，使得所有类别的概率之和为1。

**8. 编译模型**

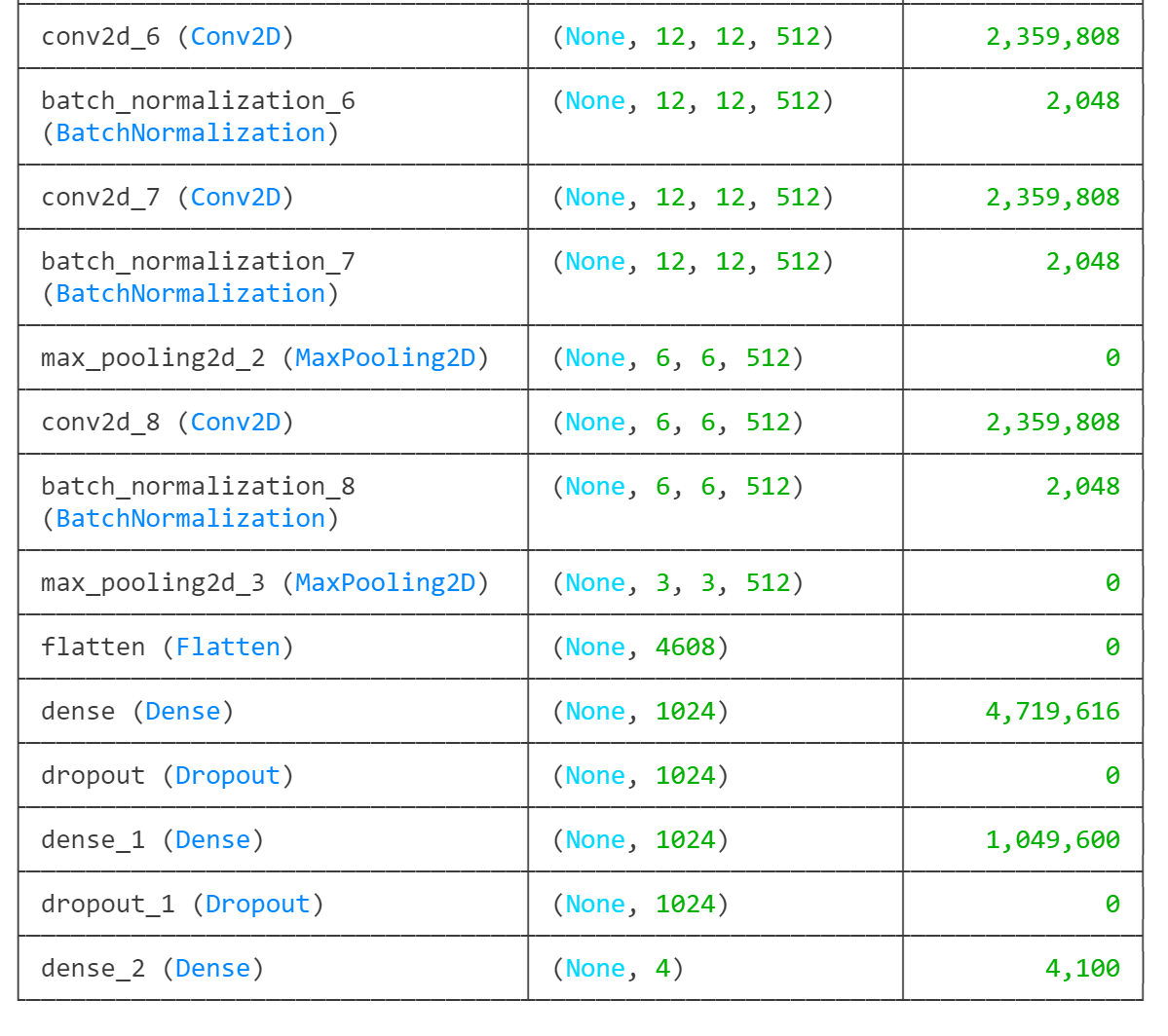
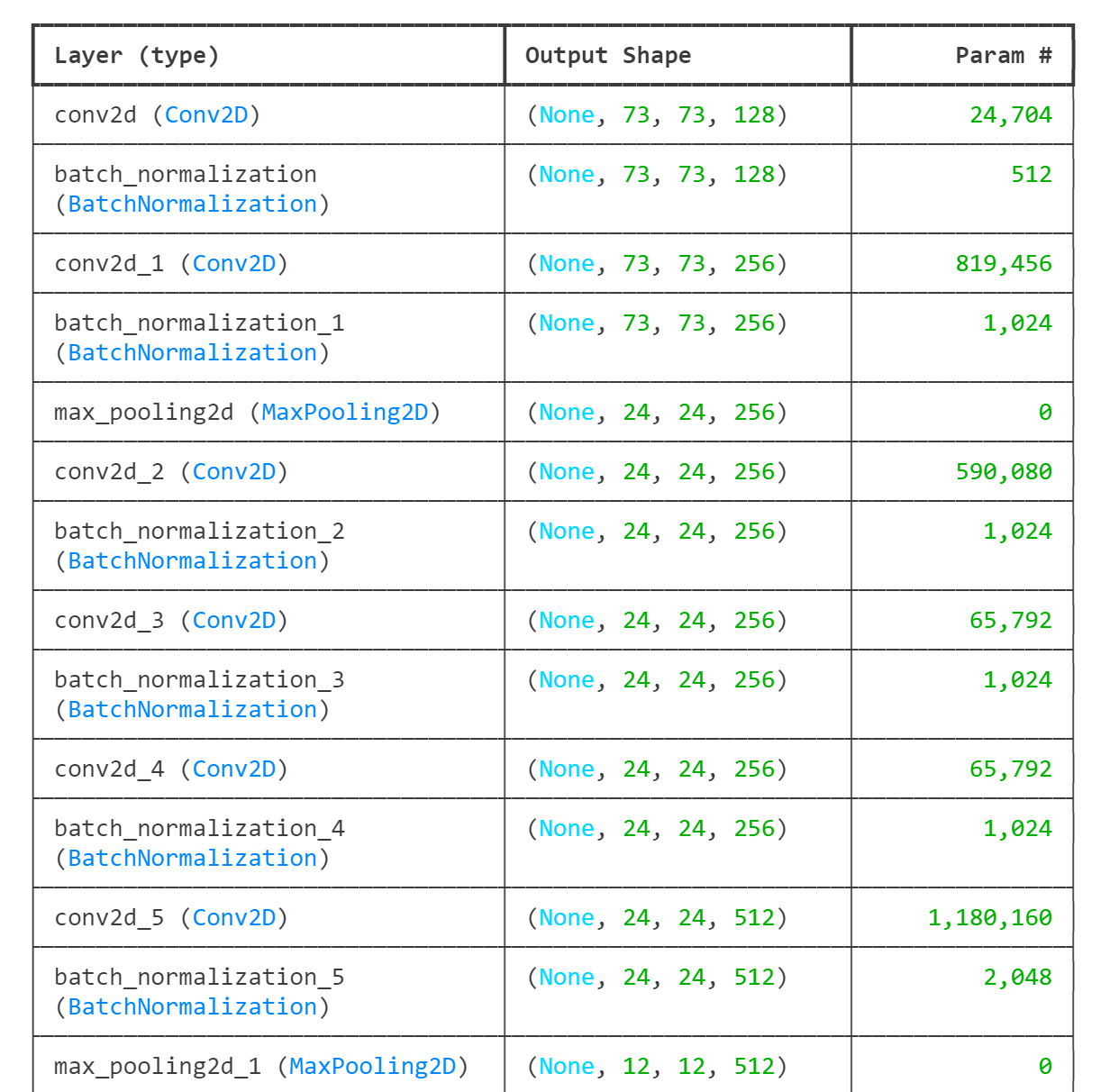
在定义完模型结构后，使用model.compile()方法来编译模型，指定优化器（如adam）、损失函数（如categorical\_crossentropy对于多分类问题）和评估指标（如accuracy）。

**9. 训练模型**

使用model.fit()方法来训练模型，传入训练数据、验证数据、批次大小、训练轮数等参数。

**10. 评估与预测**

训练完成后，可以使用model.evaluate()来评估模型在测试集上的性能，使用model.predict()来进行预测。



四、系统设计

### 1. 系统详细设计

**系统组成**：

数据处理模块：负责数据的收集、清洗、预处理和特征提取。

模型训练模块：利用处理后的图片训练机器学习模型。

模型测试模块：使用测试集对训练好的模型进行测试。

模型评估模块：根据测试结果对模型性能进行评估。

### 2. 系统流程

#### 2.1 数据处理

**数据收集**：

收集与任务相关的图像数据。

**预处理**：

对文本数据进行分词、词干化、词性标注等操作。

对图像或音频数据进行适当的缩放、裁剪、归一化等操作。

**特征提取**：

从处理后的数据中提取有用的特征，这些特征将作为模型的输入。

#### 2.2 模型训练

**模型选择**：

根据任务需求选择合适的机器学习或深度学习模型。

**参数设置**：

设置模型的超参数，如学习率、批次大小、迭代次数等。

**训练模型**：

使用训练集数据和特征对模型进行训练。

在训练过程中，使用验证集对模型进行性能监控和调优。

#### 2.3 模型测试

**准备测试集**：

从原始数据集中划分出测试集，确保测试集与训练集和验证集互斥。

**测试模型**：

使用测试集对训练好的模型进行测试。

记录模型的预测结果和真实标签，以便后续评估。

#### 2.4 模型评估

**评估指标选择**：

根据任务需求选择合适的评估指标，如准确率、召回率、F1分数、AUC等。

**计算评估指标**：

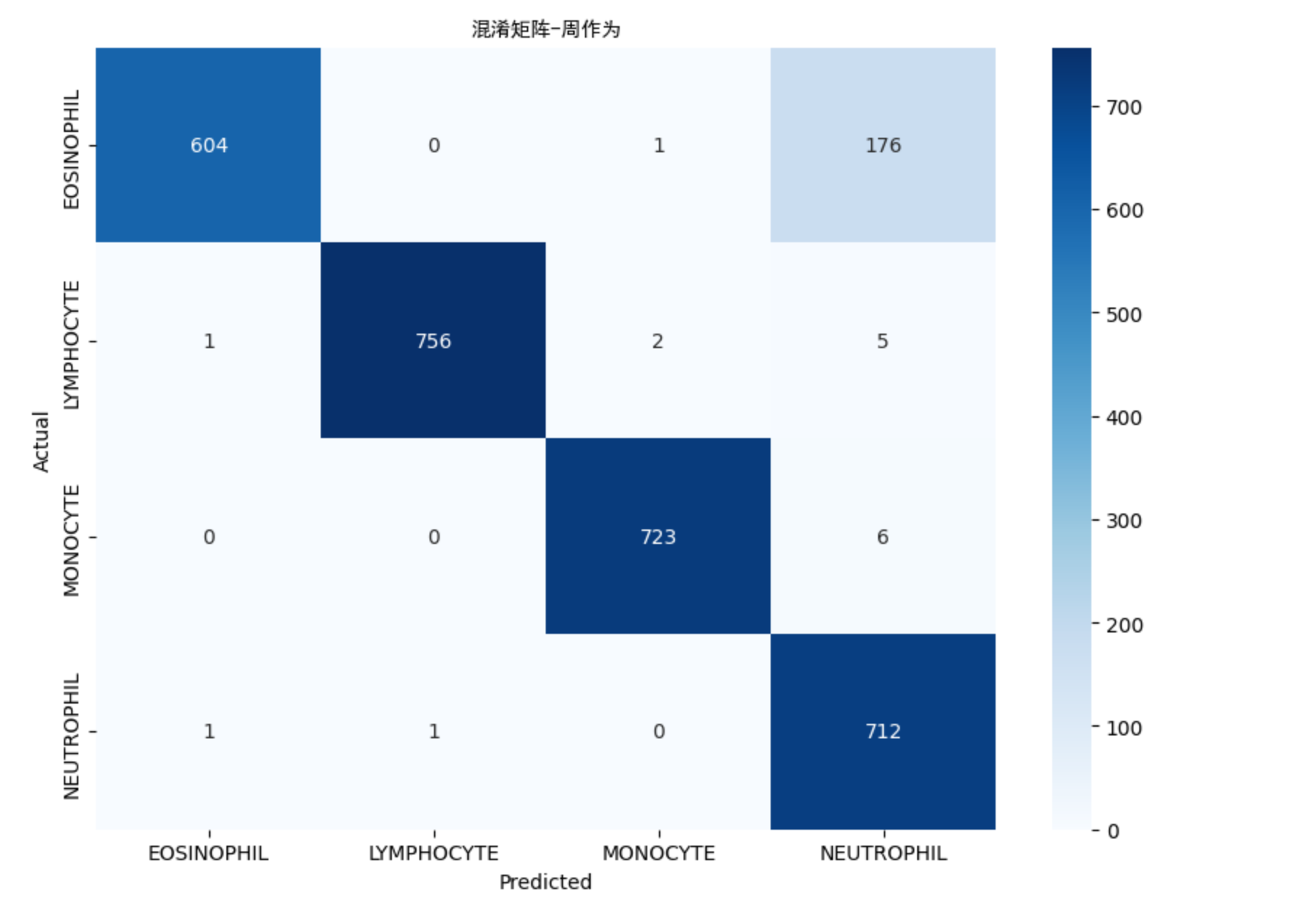
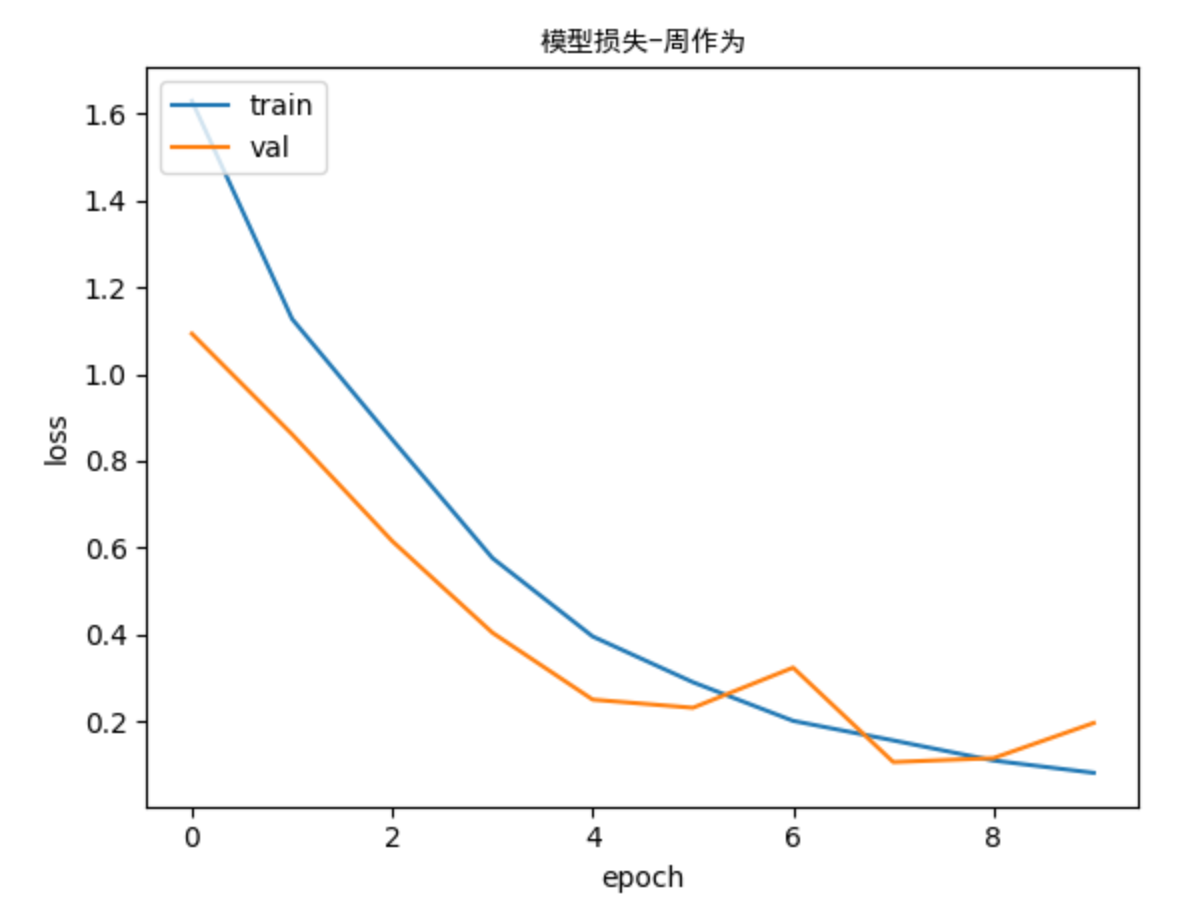
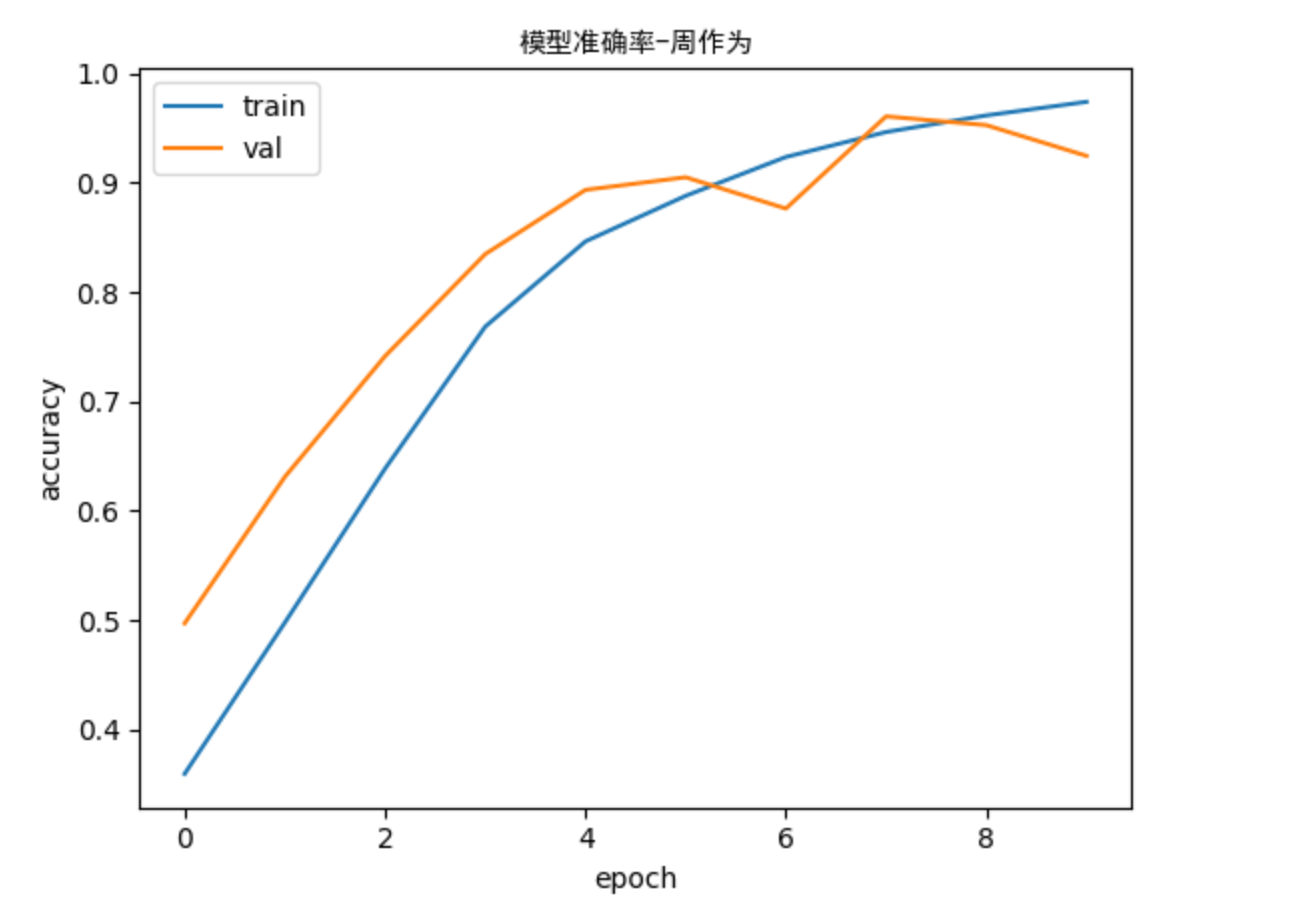
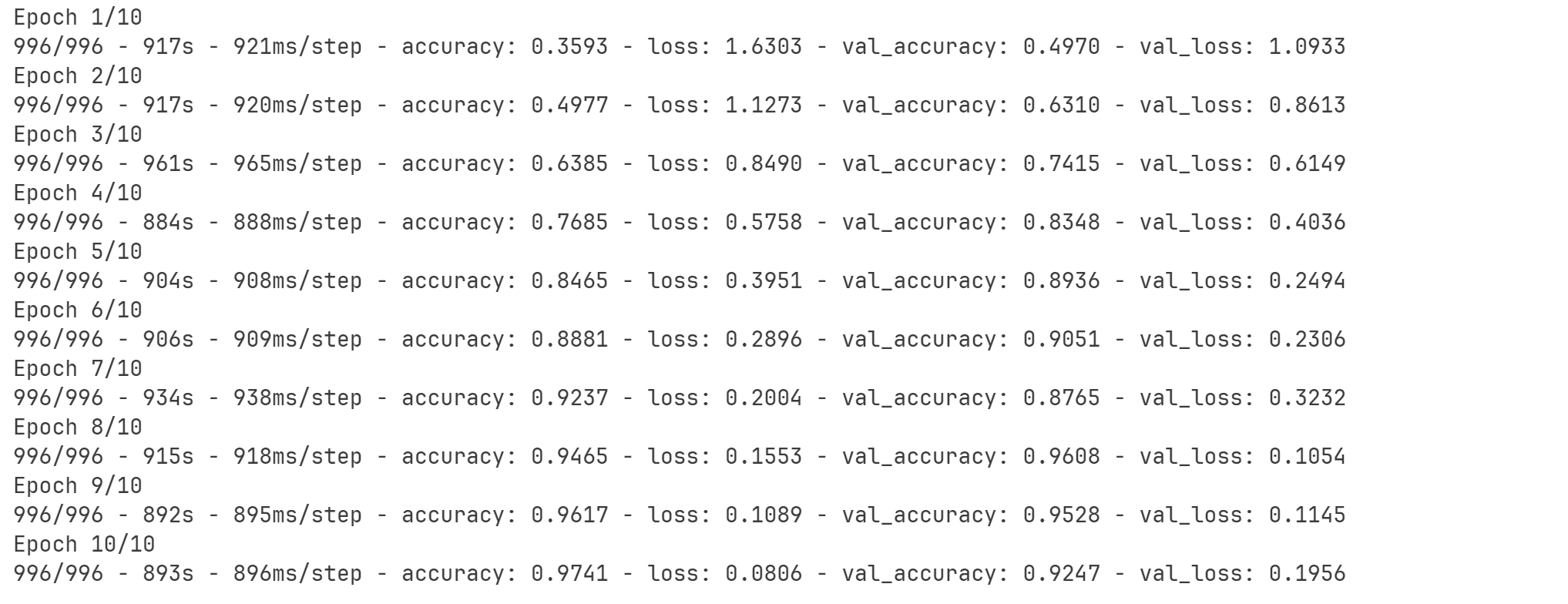
根据模型的预测结果和真实标签计算评估指标。

**性能分析**：

分析模型在不同评估指标上的表现，了解模型的优点和不足。

五．实验结果分析、对比和讨论

在训练过程中，记录每个epoch的情况，可以观察到模型的收敛情况。随着训练进行，模型在不断学习和优化。



#### 系统改进方案

1. **数据增强**：针对数据不平衡问题，可以采用数据增强技术来增加EOSINOPHIL等少数类样本的数量。这可以通过旋转、平移、缩放等操作来实现，以增加样本的多样性。
2. **模型优化**：可以尝试使用更先进的模型结构（如Transformer-based模型）或集成学习方法（如Bagging、Boosting）来进一步提高模型的性能。此外，还可以尝试使用不同的损失函数和优化算法来优化模型的训练过程。
3. **特征融合**：结合传统图像处理技术提取的特征和深度学习模型提取的特征进行融合，以提高特征的丰富性和表达能力。这可以通过将两种特征进行拼接或加权融合来实现。
4. **后处理优化**：在模型输出预测结果后，可以采用一些后处理技术来进一步提高分类的准确率。例如，可以使用阈值调整、类别重新分配等方法对预测结果进行修正。
5. **持续学习与更新**：随着新数据的不断产生和技术的不断进步，可以持续地对模型进行学习和更新，以不断提高其性能和适应性。这可以通过定期重新训练模型或在线学习等方式实现。

六．其他

**性能指标监控**：监控系统的性能指标，如响应时间、吞吐量、错误率等，以便及时发现潜在的性能问题。

**日志记录与分析**：记录系统的运行日志，并定期进行日志分析，以了解系统的运行状况和潜在问题。

**资源优化**：根据系统的负载和资源使用情况，对系统进行资源优化，如调整服务器配置、优化数据库查询等。

七．对本门课的感想、意见和建议

学习机器学习课程是一段充满挑战和收获的旅程。首先，我被机器学习的深度和广度所震撼，从基础的数据处理和算法理论，到复杂的模型训练和调优，每一步都需要深厚的数学基础和编程技能。在学习的过程中，我逐渐理解到机器学习不仅仅是一种技术，更是一种思维方式，它教会我们如何从数据中挖掘出有价值的信息，并用这些信息来解决问题。

此外，我深感机器学习的应用领域广泛，无论是医疗、金融、教育还是其他领域，都有机器学习的身影。这使我更加坚信学习机器学习的价值和意义。