**信息与软件工程学院**

**基于Python的图像分类项目开发**

**实践总结报告**

**（PBLF-2024）**

**、**

**小组成员：**

|  |  |
| --- | --- |
| **姓名** | **学号** |
| **白家乐** | **2023090913021** |
| **王天宇** | **2023090913004** |
| **周树键** | **2023091603018** |

1. 人工智能概述
   1. 人工智能的发展与现状

1.1.1 人工智能的发展分为以下几个阶段：

起步发展期（1956年-20世纪60年代初）：人工智能的概念提出后，取得了一系列突破性的研究成果，如机器定理证明和跳棋程序，掀起了人工智能发展的第一个高峰。

反思发展期（20世纪60年代-70年代初）：期望过高导致实际进展缓慢，人工智能遭遇第一次低谷。一些难以解决的问题如自然语言理解等暴露出来。

应用发展期（20世纪70年代初-80年代中）：专家系统作为重要应用形式出现，模拟人类专家解决特定领域问题的知识和经验。

低迷发展期（20世纪80年代中-90年代中）：随着应用规模扩大，专家系统的局限性逐渐显现，如知识获取困难、推理方法单一、缺乏学习能力等。

平稳发展期（20世纪90年代中-2010年）：互联网技术的发展加速了信息技术和数据资源的积累，为人工智能提供了新动力和平台。

蓬勃发展期（2011年至今）：大数据、云计算、物联网等技术进步，加上计算能力和算法改进，深度神经网络，生成式大语言模型等代表性人工智能技术飞速发展，实现了在图像识别、语音识别、自然语言处理、人机对弈、无人驾驶等领域的突破，引发了新一轮人工智能热潮。

1.1.2 人工智能的发展现状：

现今，专用人工智能领域取得了显著进展。专用人工智能系统专注于特定任务，如图像识别、语音识别、自然语言处理等，在这些领域取得了令人瞩目的成就。

目前，各类人工智能大模型处于迅猛发展之中，全球众多高科技企业纷纷投身人工智能大模型建设。人工智能大模型的出现，为通用人工智能的实现打开了新的想象空间。比如，基于人工智能大语言模型的聊天机器人能够实现高质量的信息整合、翻译和简单问题求解与规划。

人工智能的创新和创业活动也非常活跃。越来越多的公司和创业者投身于人工智能领域，推动着技术的不断进步和商业应用的不断拓展。

同时，政府和学术界也在积极支持人工智能的发展，投资于研究项目、建立实验室和推动政策制定。

人工智能发展过程中也面临一些挑战和问题。例如，数据隐私和安全、算法偏见和歧视、人机共生的伦理问题等，都需要我们认真思考和解决。

* 1. 人工智能的应用

人工智能在当今社会的应用已经涵盖了几乎所有的行业和领域，其应用范围越来越广泛，影响力越来越深远。以下是一些人工智能在当今社会中的主要应用领域：

1. 医疗保健：如疾病诊断、影像识别、个性化治疗方案设计等。
2. 金融服务：如风险管理、欺诈检测、交易预测等方面。
3. 零售和电子商务：如个性化推荐、精准营销、库存管理等方面。
4. 智能交通：如智能交通管理、智能驾驶、交通预测等。
5. 智能制造：如预测性维护、智能生产调度、机器人自动化等方面。
6. 教育培训：如个性化教学、智能辅导等方面。
7. 智能家居：如智能家电控制、智能安防监控、智能环境感知等方面。
   1. 人工智能技术与分类
8. 机器学习：机器学习是一种让计算机通过数据自动学习的方法，它可以分为监督学习、无监督学习和强化学习。
9. 深度学习：深度学习是机器学习的一个分支，它通过多层神经网络来学习复杂的模式和特征。
10. 自然语言处理：自然语言处理涉及到文本的理解和处理，其中常见的任务包括文本分类、机器翻译、命名实体识别和语义分析等。
11. 计算机视觉：计算机视觉是研究如何让计算机“看懂”图像和视频的领域，包括图像分类、目标检测、人脸识别和图像生成等任务。
12. 自动化和机器人技术：自动化和机器人技术涉及到利用机器人和自动化系统来执行任务，包括自动化流程和机器人技术两个方面。
13. 图像分类任务概述

2.1图像分类应用与技术概要

2.1.1 图像分类技术的应用

1. 物体识别和分类：通过训练深度学习模型，可以将分类到预定义的类别中。这在自动驾驶、安防监控、医学影像分析等领域有广泛应用。
2. 人脸识别：利用深度学习可以进行人脸检测，可用于身份验证、安防监控、社交媒体应用等。
3. 图像搜索：基于图像内容的相似性，可以建立图像搜索引擎，用户可以通过上传图像来查找相似或相关的图像。
4. 医学影像分析：利用深度学习技术可以对医学影像进行分析和诊断，如癌症检测、病理学分析等。
5. 农业领域：图像分类可以帮助农民监测作物健康状况、检测病虫害，并进行精准农业管理。
6. 图像内容审核：用于社交媒体、在线论坛等平台，通过自动识别图像内容来进行审核，以过滤不良内容。
7. 品质控制：在制造业中，利用图像分类技术可以检测产品的缺陷或质量问题，帮助提高生产线的效率和产品质量。
8. 文档识别与分类：通过图像分类技术，可以对文档中的文字、图表等内容进行识别和分类，用于自动化文档管理、信息提取等任务。
9. 地理信息系统（GIS）：在地图制作和地理信息系统中，利用图像分类技术可以识别地物类型，如道路、建筑物、植被等，从而帮助更新地图信息和进行城市规划。
10. 环境监测：利用图像分类技术可以监测自然环境中的变化，如森林覆盖、水体变化等，用于环境保护和资源管理。

2.1.2 图像分类的技术概要

图像分类技术概要大概可以分为一下几个方面：

1. 图像特征提取。常用的方法有使用颜色直方图、提取图像的纹理特征、提取图像的形状特征、使用卷积神经网络（CNN）提取图像的高级特征。
2. 分类器。它是图像分类任务中的核心组件，它用于对特征向量进行分类。常见的分类器包括：SVM、随机森林、神经网络。
3. 损失函数。用于评估分类器的性能。
4. 算法优化。优化算法用于训练分类器。如使用梯度下降法来训练神经网络。

2.2图像分类技术

1. 使用了百度飞桨 PaddlePaddle 深度学习开发套件。

2. 使用 PaddleHub 序训练模型库。

3. 使用 resnet\_v2\_50\_imagenet 预训练模型。

4. 配合 Finetune API 基于大规模预训练模型快速完成迁移训练。

5. 使用 BaseCVDataset 库自定义数据集，根据项目需要对模型进行食品五分类迁移训练。

2.3 图像分类实现环境

项目基于Linux操作系统，Python编程语言，运行与百度飞桨线上编程平台。

采用线下编码线上运行的方式。线下使用Visual Studio Code代码编辑器，配合 python venv 虚拟环境管理器，成功完成了对面包、甜品、蛋类、肉类、面条的五分类任务。

程序模块化编写，大大方便了后期的维护和更改。

使用 Github 代码托管平台管理项目代码，使代码可以第一时间在组内得到同步。（地址：<https://github.com/syxb2/image_classification_semester2prj>）

1. 个人任务实现

3.1 个人任务概述

白家乐：制作train/validate/test各个list以及labellist，制作主程序，代码整理。

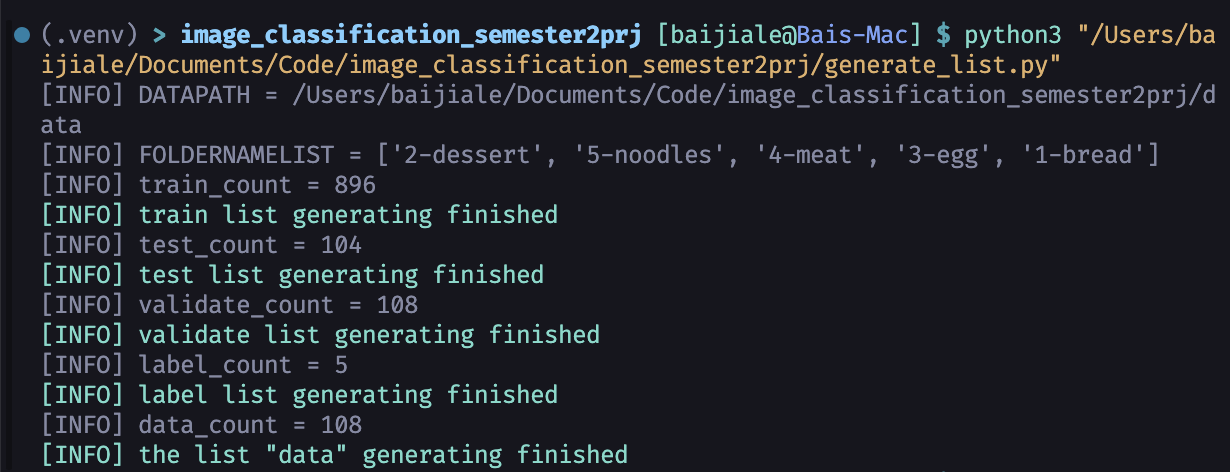
周树建：完成主程序finetune以及计算各个指标、混淆矩阵和统计图示展示。

王天宇：数据集增强。

3.2 白家乐同学个人任务实现

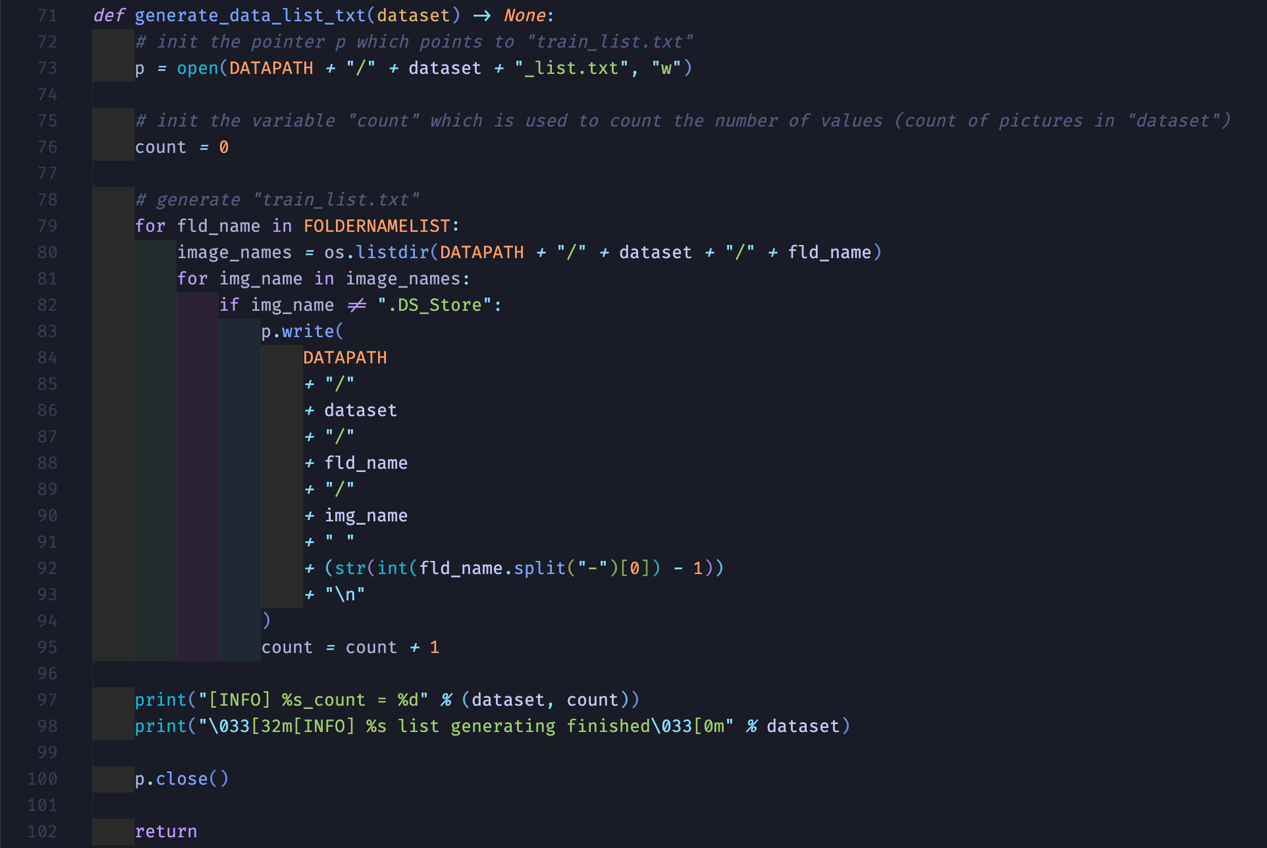
实现模块化根据数据集制作finetune自定数据集所需要的train/validate/test各个list以及label list，实现生成列表程序的模块化设计，便于后续的更改与维护。

3.2.1 运行结果：（见图1）

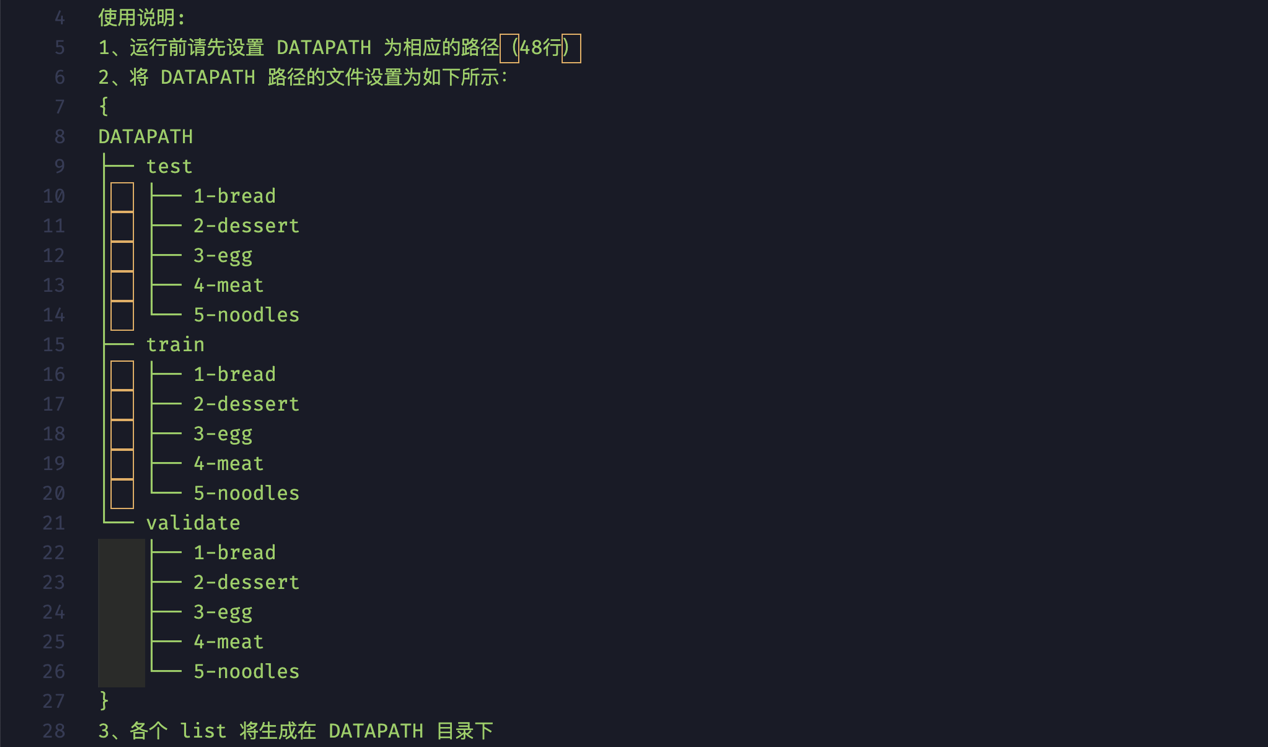
图1

3.2.1 模块化代码示例：

这里是生成train/validate/test 三个列表的函数：（见图2）

****图2

3.2.3 整个模块的使用说明：



3.3 周树建同学个人任务实现

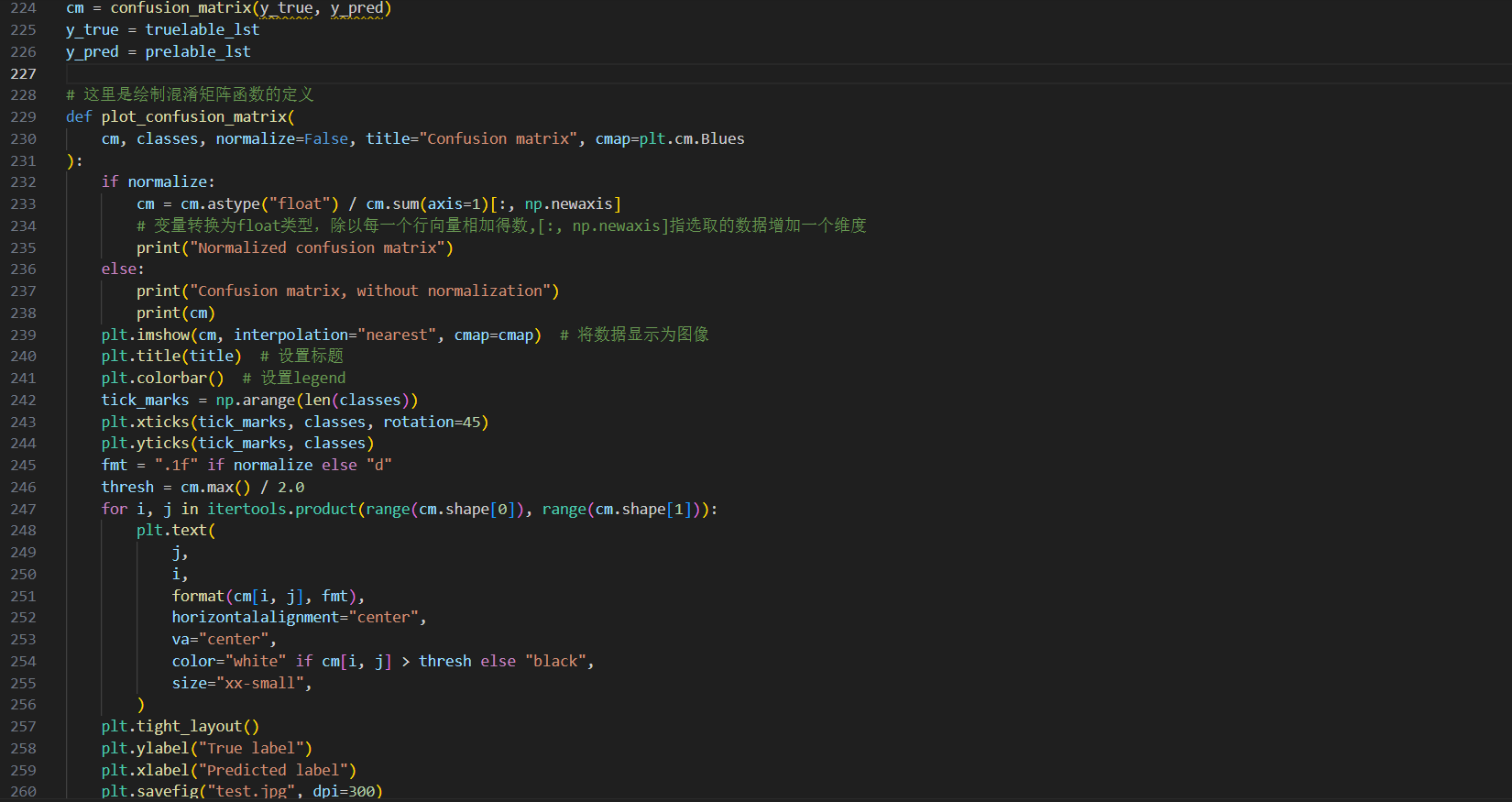
3.3.1 模块化代码示例：

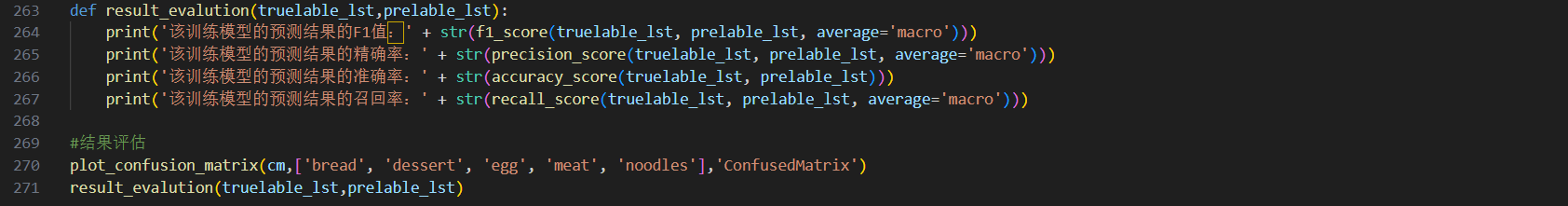
将验证集数据从文件读出，生成验证图片地址列表，输出模型预测结果，将真实标签及预测标签列表按照字典生成对应的混淆矩阵



3.3.2 模块化代码示例：

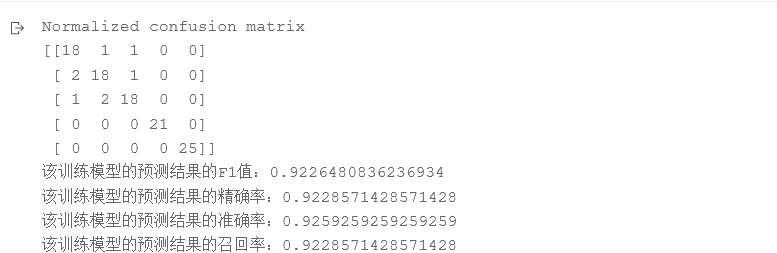
绘制混淆矩阵，并计算结果的精确率，准确率，召回率及F1值

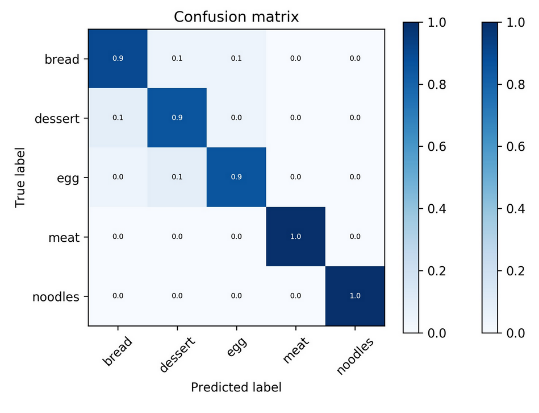




3.3.3 模型预测结果输出

将验证集数据从文件读出，生成验证图片地址列表，输出模型预测结果，将真实标签及预测标签列表按照字典生成对应的混淆矩阵

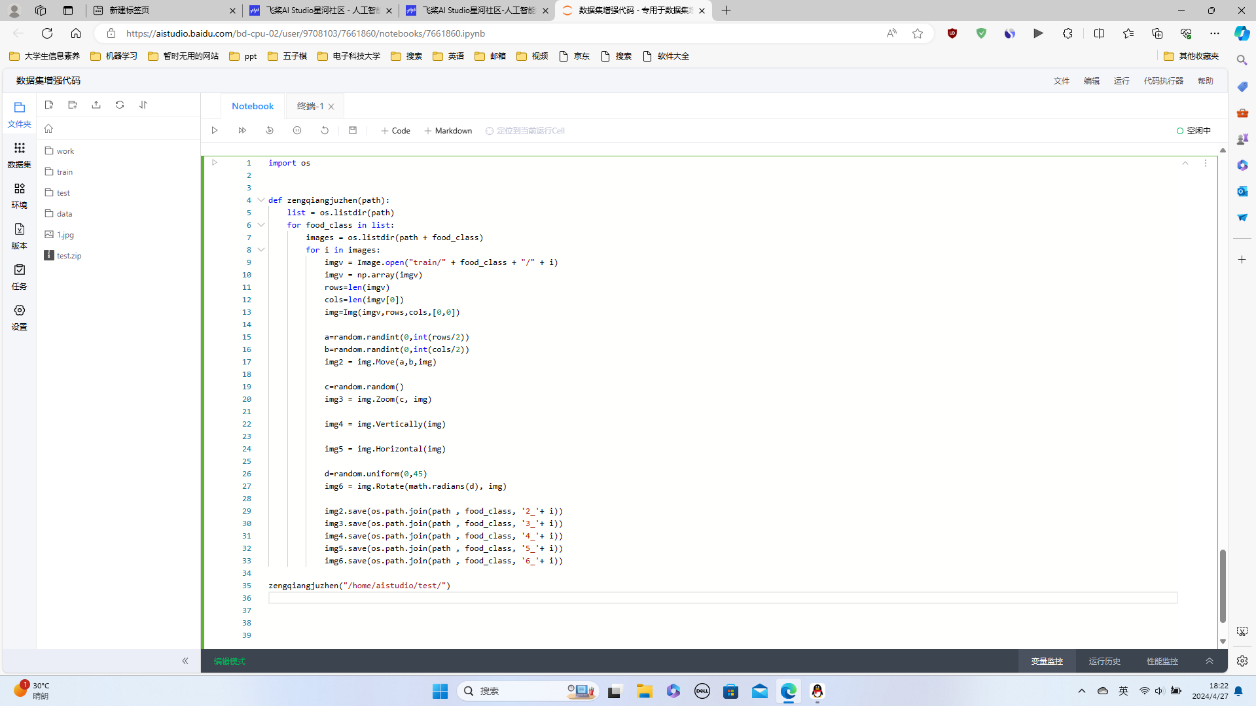




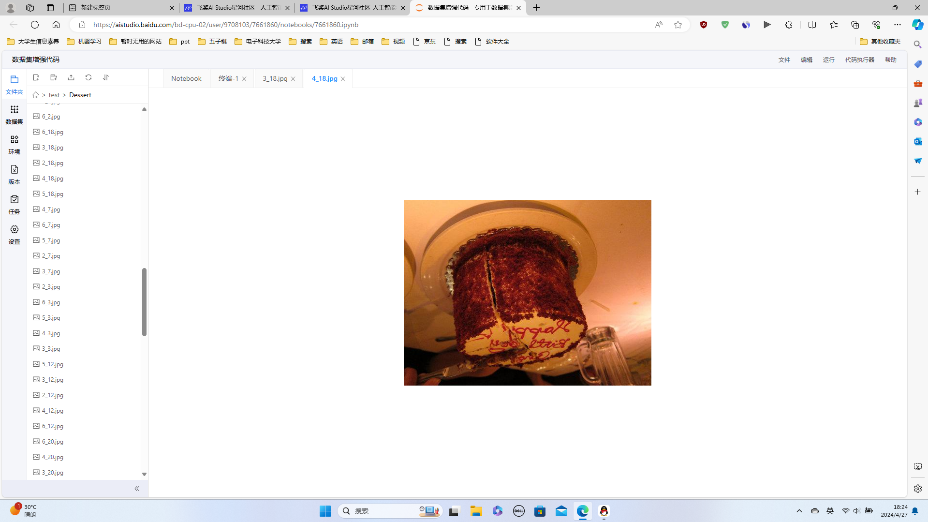
3.4 王天宇同学个人任务实现

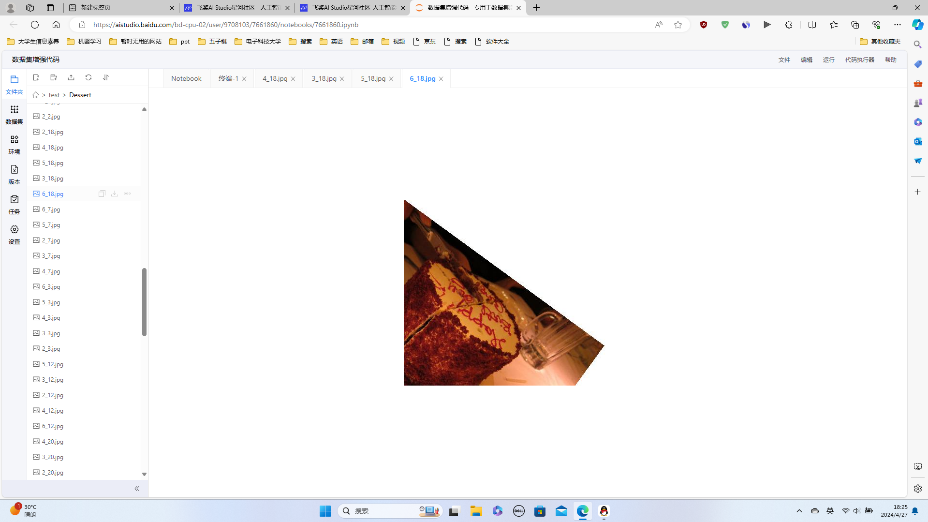
3.4.1代码模块实例

将原有数据集的路径输入，经过逐层搜索将图片进行平移，旋转，缩放，反转，颠倒等操作扩充数据集，减少图片噪声，避免模型过拟合，



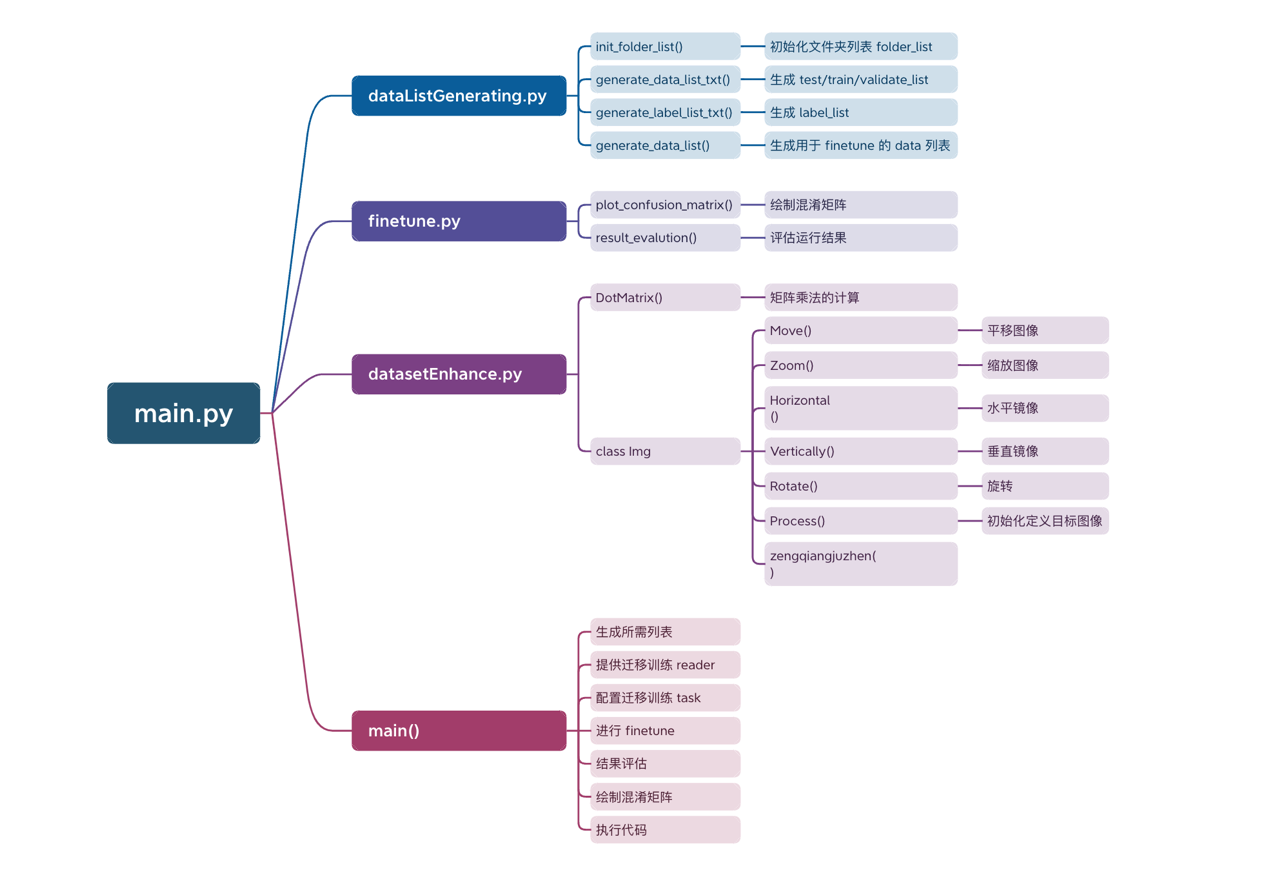
3.4.2数据集增强个例展示





1. 迁移训练项目实现分析

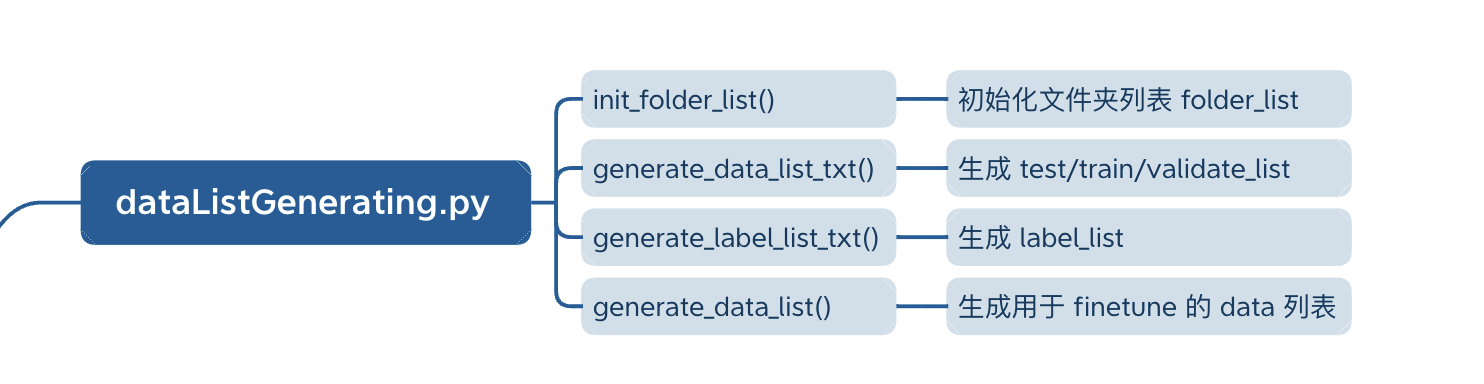
4.1任务架构



4.2任务模块具体分析及实现

4.2.1 dataListGenerating.py

* + - 1. 结构图



4.2.1.2 具体实现

这里以生成数据列表的函数为例：（见表1）

def generate\_data\_list\_txt(dataset) -> None:

# init the pointer p which points to "train\_list.txt"

p = open(DATAPATH + "/" + dataset + "\_list.txt", "w")

# init the variable "count" which is used to count the number of values (count of pictures in "dataset")

count = 0

# generate "train\_list.txt"

for fld\_name in FOLDERNAMELIST:

image\_names = os.listdir(DATAPATH + "/" + dataset + "/" + fld\_name)

for img\_name in image\_names:

if img\_name != ".DS\_Store":

p.write(

DATAPATH

+ "/"

+ dataset

+ "/"

+ fld\_name

+ "/"

+ img\_name

+ " "

+ (str(int(fld\_name.split("-")[0]) - 1))

+ "\n"

)

count = count + 1

print("[INFO] %s\_count = %d" % (dataset, count))

print("\033[32m[INFO] %s list generating finished\033[0m" % dataset)

p.close()

return

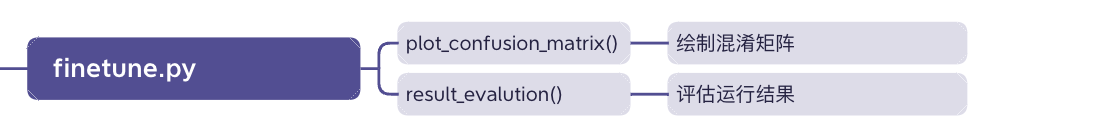
表1

效果：实现模块化根据数据集制作finetune自定数据集所需要的train/validate/test 各个list以及label list，实现生成 finetune 所需要的 data 列表

**（你们自己分解出的每个模块（有的时候就是一个功能性的函数）的功能，整体流程图，每个模块的具体实现（含核心代码），实现的步骤，描述实现效果，不作字数要求）**

4.2.2 finetune.py and main.py

4.2.2.1 结构图

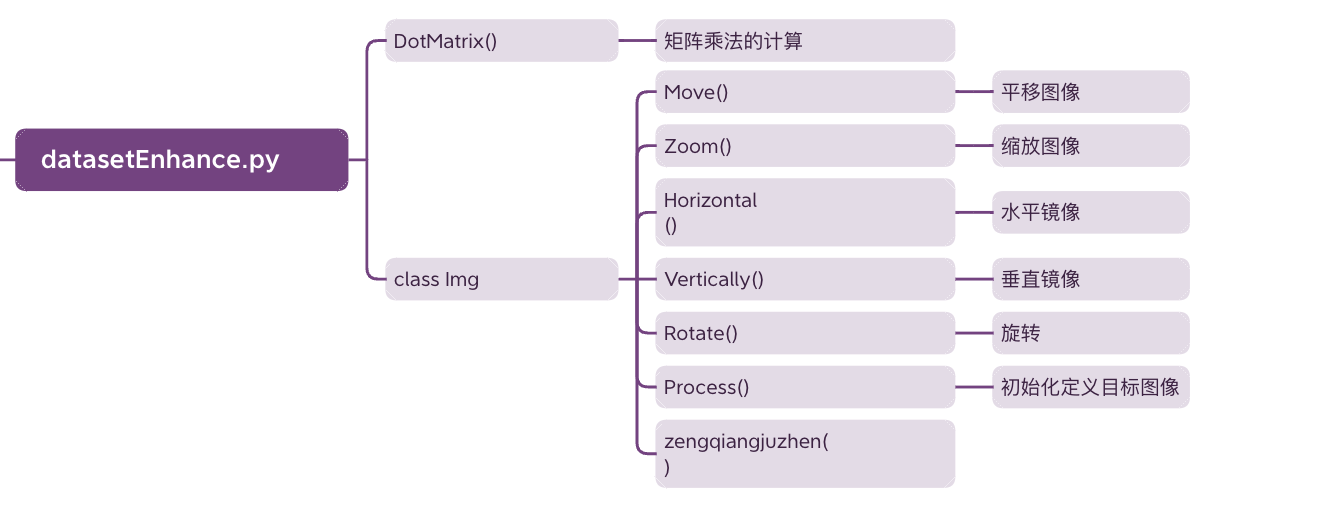




TODO

4.2.3 datasetEnhance.py

4.2.3.1 结构图



TODO

4.3 项目实现情况

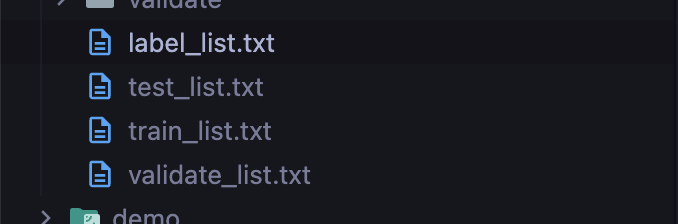
4.3.1 分工实现

4.3.1.1 dataListGenerate.py

实现：白家乐

实现模块化根据数据集制作finetune自定数据集所需要的train/validate/test 各个list以及label list，实现生成 finetune 所需要的 data 列表，实现生成列表程序的模块化设计，便于后续的更改与维护。

函数代码及实现步骤见 4.2.1.2

效果截图：



**（你们自己分解出的每个模块（有的时候就是一个功能性的函数）的功能，整体流程图，每个模块的具体实现（含核心代码），实现的步骤，描述实现效果，不作字数要求）**

4.3.1.2 finetune.py and main.py

实现：周树建

TODO

4.3.1.3 datasetEnhance.py

实现：王天宇

TODO

**团队任务各位同学分工的情况和实现效果截图**

4.3.2 项目整体完成情况

完美实现利用预训练模型并使用自定义数据集finetune完成对食物的五分类

**（最终团队任务完成情况，指标点计算和指标点的图示展示等等）**

1. 复杂工程问题分析
   1. 运行速度太慢问题及解决方案

问题描述：数据集增强的过程以及 finetune 迁移训练的过程的运行速度太慢。

解决方案与解决原理：减小部分数据集的图片大小，降低模型训练的性能压力。

* 1. 项目所需 PaddlePaddle 版本太老问题及解决方案。

问题描述：项目以来的 PaddlePaddle版本太老，不支持 ARM 架构芯片，组内使用 ARM 架构芯片的成员无法本地使用 PaddlePaddle 库。

问题解决：重新分配组内任务，将需要使用 Paddle 库的任务分配给组内使用 x86 架构芯片的成员。

查阅的资料：百度飞桨官方 Github 文档。

1. 团队协作和人员实现展示

6.1 团队协作

6.1.1 团队协作心得

通过完成这次的小组合作的图像分类任务，我们有以下心得：

要有明确的分工：在开始项目之前，明确每个成员的角色和任务。例如，可以有一个负责数据收集和预处理，一个负责任务所需各种列表的制作，以及一个负责迁移训练和结果评估。

要定期沟通：保持团队之间的定期沟通非常重要。可以通过定期会议、在线聊天等方式进行沟通，分享进展、讨论问题并制定解决方案。

要积极分享资源和知识：团队成员应该积极分享有用的资源和知识，包括相关文献、教程、代码示例等。这有助于提高整个团队的技术水平和工作效率。

要及时共享代码：采用版本控制系统（Git和Github）管理代码，并确保团队成员能够轻松地共享和合作编写代码。这样可以避免冲突，并能够追踪项目的进展和历史修改记录。

要持续反馈和改进：团队成员之间应该保持开放的沟通和反馈机制，及时分享对项目的想法、建议和改进意见。这有助于及时解决问题并不断提高项目的质量和效率。

6.2 小组成员得分

|  |  |  |
| --- | --- | --- |
| 学号 | 姓名 | 得分（100分制） |
| 2023090913021 | 白家乐 |  |
| 2023090913004 | 王天宇 |  |
| 2023091603018 | 周树键 |  |

6.3 项目总结

在这次 PBLF 的课程中，我们组内三人合作完成了对食品图像的五分类任务，通过这次的实践，我们有以下心得：

通过这次项目的实践，我们认识到团队协作在项目中至关重要。这次项目展示了团队协作的重要性。每个团队成员在项目中发挥了关键作用，共同努力才使得项目得以顺利完成。

第二，我们认识到数据集的重要性。在图像分类任务中，拥有高质量的数据是至关重要的。通过整理食品图像数据集，使得模型能够更准确地进行分类。同时，数据的多样性和代表性也对模型的性能有着重要影响。

第三，这次项目不仅是任务完成，也是经验的积累过程。团队成员通过实践掌握了图像分类任务的流程和技术，积累了宝贵的经验和知识，为未来类似项目的开展奠定了坚实的基础。

第四，完成这次项目之后，团队成员应该继续保持学习的态度，关注领域内的最新进展和技术，不断改进和完善自己的能力。只有不断学习和提升，才能在未来的项目中取得更好的成绩。

第五，在项目过程中可能会遇到各种挑战和问题，例如运行时间太慢等。团队需要有耐心和毅力去解决这些问题，可以通过团队讨论、在线资源查找、尝试不同的方法等方式来克服困难。

第六，代码的注释和可读性十分重要。这有助于团队成员之间更好地理解和沟通，提高代码可维护性。

第七，项目管理和时间规划：良好的项目管理和时间规划对于项目的顺利进行至关重要。团队可以使用项目管理工具来安排任务和跟踪进度，确保项目能够按时完成。

最后，项目完成后，团队可以进行反思和总结，分析项目的成功因素和不足之处，并提出改进意见。这有助于团队从经验中吸取教训，为未来的项目提供借鉴。