

软件工程综合实训项目任务书

* 一、项目背景

随着大数据与人工智能技术的迅猛发展，软件工程项目从传统的信息系统逐渐向数据驱动和智能化方向转型。本次综合实训旨在让学生将所学的软件工程知识与人工智能技术进行综合运用，完成一个小型的智能应用系统，进一步巩固学生的专业实践能力、团队协作能力，并为将来就业或科研打下坚实基础。

* 二、实训目的

- ① 综合应用专业知识：以需求分析、系统设计、开发、测试、部署的标准软件工程流程为指导，将软件工程与人工智能理论知识在项目实践中融会贯通，培养学生的工程实践与研发能力。
- ② 锻炼团队协作与项目管理能力：通过分工协作、问题沟通、项目管理与推进、成果总结与展示等环节，训练学生的团队协作、沟通表达与项目管理能力。
- ③ 培养人工智能应用与创新素养：通过深入实践深度学习与计算机视觉等人工智能技术的基础应用与部署，了解算法原型到可用服务的落地过程；在选题、功能设计、解决方案探索中鼓励创新，培养学生的创新素养。

* 三、项目选题

本项目提供三个选题，前两个为指定选题，第三个为自由选题。项目以小组为单位进行，每个小组需要从以下项目选题中选择一个，按要求认真完成实训任务，并按时提交项目成果和参加答辩。

选题一：智能垃圾分类系统

（1）任务简介

垃圾分类（garbage classification）图像数据集包含了 6 种常见生活垃圾类别的图像，分别为纸板 (cardboard)、玻璃 (glass)、金属 (metal)、纸(paper)、塑料 (plastic)、一般垃圾 (trash)，图像总数为 2534。请基于该垃圾分类数据集(见附件)，完成数据预处理、深度学习模型的设计、训练与性能评估、以及系统的设计与开发，最终得到一个智能垃圾分类系统，支持上传图片进行垃圾分类。

（2）项目要点

- 数据预处理：如训练集、验证集、测试集的划分，数据增强策略等；
- 模型设计：如采用自定义模型或预训练模型（VGG、ResNet、Transformer等）；
- 模型训练：如损失函数、优化器、学习率、训练时长等调试；
- 性能评估：包括模型效果（如分类精度, 准确度, 召回率, F1分值）以及模型复杂度（模型大小、计算量、推理时间等）；
- 系统设计与开发：Web或桌面应用均可，前端上传及显示图片，后端运行模型推理及返回结果。

选题二：智能行人检测与分割系统

（1）任务简介

Penn-Fudan 行人检测与分割数据集是由宾夕法尼亚大学的研究人员联合复旦大学团队收集整理，用于行人检测(Pedestrian Detection)与行人分割(Pedestrian Segmentation) 任务的公开数据集。Penn-Fudan数据集共有 170张图像，包含 345 个带有检测框标注和分割标注的行人实例，可用于实现行人检测和行人分割。请基于该数据集（见附件），完成数据预处理、深度学习模型的设计、训练与性能评估、以及系统的设计与开发，最终得到一个智能行人检测与分割系统，支持上传图片并检测与分割图片中的行人。

（2）项目要点

- 数据预处理：如训练集、验证集、测试集的划分，数据增强策略等；
- 模型设计：如采用自定义模型或预训练模型（如Mask R-CNN等）；
- 模型训练：如损失函数、优化器、学习率、训练时长等调试；

- 性能评估：包括模型效果（如mAP、IoU等）以及模型复杂度（模型大小、计算量、推理时间等）；
- 系统设计与开发：Web或桌面应用均可，前端上传及显示图片，后端运行模型推理及返回结果。

选题三：自定义智能系统（自由选题）

（1）任务简介

结合所学专业知识、自身兴趣以及实际问题需求，围绕智能系统设计及开发主题，自主提出选题，包括需求分析与问题定义、数据获取与预处理、模型设计、训练与评价、以及系统设计与开发。其中，数据获取可以通过自行采集数据构建数据集，也可以选择网上公开的数据集(如Kaggle、Github、HuggingFace等平台)。选题要有一定的复杂性和综合性，但应在能力和算力资源范围内合理规划，鼓励结合实际问题。数据收集需合法合规，避免侵犯隐私和版权。

（2）项目要点：

- 数据获取及预处理：如数据采集的方式及流程、训练集、验证集、测试集的划分，数据增强策略等；
- 模型设计：如采用自定义模型或预训练模型；
- 模型训练：如损失函数、优化器、学习率、训练时长等调试；
- 性能评估：包括模型效果以及模型复杂度；
- 系统设计与开发：Web或桌面应用均可，前端上传及显示图片，后端运行模型推理及返回结果。

✧ 四、实训时间节点

- ① 组队与选题（寒假前）：自由组队，每组3～5人，含1名组长（组长有适当加分）；
- ② 项目开展（寒假中，尽早启动）：组长协调项目分工协作、进度推进、资源协调、问题反馈以及成果提交，全体组员认真完成实训任务；
- ③ 答辩与成果提交（寒假结束第一周）：以小组为单位，参加项目成果展示与答辩，并提交项目成果。

❖ 五、成果形式与提交要求

- ① 项目报告：包括项目背景介绍、数据获取（如有）与预处理流程、模型与系统设计、实现过程说明、测试结果与分析、项目总结与展望，参考文献等。
要求报告内容完整、结构合理、逻辑清晰、格式规范，不少于 5000 字；
- ② 项目代码：包含可编译、可运行的完整项目，代码结构清晰、注释规范，README文件简明说明项目启动方式；
- ③ PPT演示文稿：主要展示项目背景、需求概述、技术架构、关键功能/算法、效果演示、项目心得与展望；页面简洁，逻辑清晰，突出重点，长度合理；
- ④ 演示视频：录制一段操作演示视频，方便评审老师快速了解项目成果。

❖ 六、考核与评分标准

- ① 功能实现（30%）：系统功能完备度、可用性、稳定性；是否满足需求，能否在演示中流畅运行；
- ② 技术深度与创新性（30%）：算法模型或技术框架选型是否合理，是否有改进或创新，工作量大小；数据处理、参数调优、模型效果等方面的深度；
- ③ 报告质量（20%）：报告的逻辑性、完整度、可读性与格式规范性；
- ④ 团队协作与项目管理（10%）：角色分工与配合度、团队沟通效率、项目管理与进度推进情况；
- ⑤ 答辩与展示（10%）：PPT演示逻辑清晰度、表达流畅度、美观度、完整度和时间把握；现场Demo效果，回答问题的逻辑性、准确度与表现力；

❖ 七、参考资料

[1] 垃圾分类数据集：

<https://www.kaggle.com/datasets/asdasdasdasdas/garbage-classification?resource=download>

[2] Penn-Fudan数据集：https://www.cis.upenn.edu/~jshi/ped_html/

[3] PyTorch行人检测与分割教程：

https://pytorch.org/tutorials/intermediate/torchvision_tutorial.html

[4] GitHub：<https://github.com/>

[5] Kaggle: <https://www.kaggle.com/>

[6] HuggingFace: <https://huggingface.co/>

[7] PyTorch: <https://pytorch.org/>