

人工智能原理与技术课程设计内容要求与评分细则

1. 逻辑推理模块：八皇后问题

1.1. 实验背景

逻辑编程是一种编程典范，它设置答案须匹配的规则来解决问题，而非设置步骤来解决问题。过程是“事实+规则=结果”。人工智能的发展与逻辑编程的发展是一个相辅相成的过程，早期的人工智能以规则和逻辑推理作为主要研究方向，这在逻辑编程的发展中发挥了重要的影响，另外更好更快的逻辑编程也推动了人工智能的发展，例如专家系统、知识图谱和自动定理证明。

Python 是一种解释型、面向对象、动态数据类型的高级程序设计语言。在数据驱动学习时代，Python 的崛起已经是一个不争的事实，并且成为人工智能算法的第一语言。在本次实验中，我们学习将 Python 应用于逻辑编程，并尝试自主撰写逻辑规则解决八皇后问题。

1.2. 实验内容

八皇后问题：如何能够在 8×8 的国际象棋棋盘上放置八个皇后，使得任何一个皇后都无法直接吃掉其他的皇后？为了达到此目的，任两个皇后都不能处于同一条横行、纵行或斜线上。如下图所示：



1.3. 实验要求

- a). 基本掌握逻辑编程的思想，了解逻辑编程与命令式编程的区别
- b). 能够依据给定的事实以及规则编写代码，解决逻辑约束问题（CLP）

1.4. 实验环境

使用 Python 语言，Kanren 库进行逻辑编程。

1.5. 参考资料

Python:

<https://www.runoob.com/python3/python3-tutorial.html>

kanren:

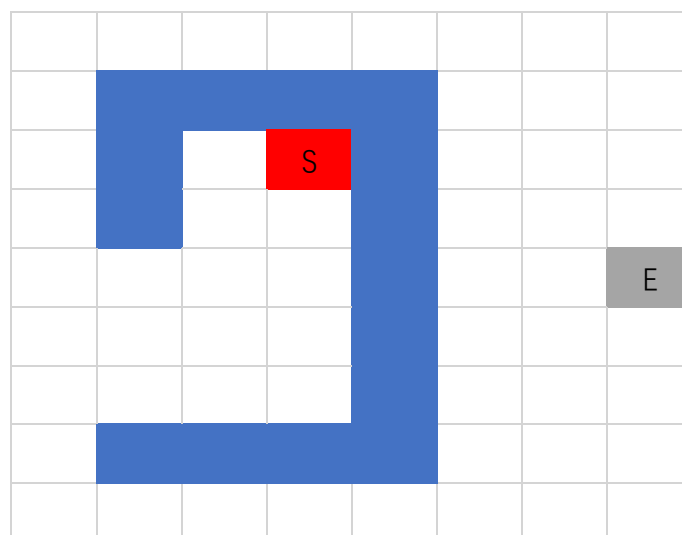
<https://github.com/logpy/logpy>

2. 搜索模块：机器人避障寻径问题

2.1 实验内容

机器人避障寻径是机器人领域的一项重要技术，该技术要求实现机器人在复杂环境中的自主导航。在现实生活中，机器人的自主导航需要考虑各种因素，包括环境的复杂性、障碍物的位置和形状、机器人的运动能力等。如何解决这些问题是机器人避障寻径任务的研究难点。基于规则的传统路径规划方法，如 A*算法、Dijkstra 算法等，已经在机器人路径规划中得到广泛应用，并能够在大部分场景中实现较高的准确度。这些算法的核心思想是通过搜索算法在地图中找到最优的路径。这些算法具有计算复杂度低、实现简单等优点。

在本实验中，我们给定如下的机器人避障寻径场景图：



其中 S 为机器人寻径的起点, E 为寻径的终点, 蓝色区域为无法通过的障碍。要求机器人从 S 点出发, 绕开障碍, 到达 E 点。本实验需要参与者自行设计核心算法规则完成机器人从起点到终点的最短路径搜索。

2.2 实验要求

- 1) 已提供场景图的的矩阵表示, 以及机器人起始点和终止点的坐标
- 2) 要求找到从起始点到终止点的最短路径
- 3) 禁止采用 hard code 的方式
- 4) 可视化最短路径图搜索过程

2.3 选做内容(加分项)

1) 自主设计更高难度场景图，并结合强化学习、深度强化学习等算法解决问题

2) 分析所设计算法的局限性，及改进点

3) 最短路径查找过程的动态化展示

4) 提出第二种完全不同的解决方案

2.4 实验环境

建议采用 python 来实现目标算法。允许调用基于 Python 的 Pandas、Numpy、Sklearn 等算法库。Pytorch 等深度学习框架也是被鼓励使用的。

2.5 注意事项

在做选做过程中，如果用到深度学习模型，需要搭建模型运行环境。为保证深度学习模型的正常运行，不同 python 库之间版本需要正确匹配。建议使用 anaconda 自动管理各种 python 库。

2.6 参考资料

Python:

<https://www.runoob.com/python3/python3-tutorial.html>

Numpy:

<https://www.numpy.org>

Sklearn:

<https://scikit-learn.org>

Pytorch:

<https://pytorch.org/>

参考文献:

A note on two problems in connexion with graphs

3. 深度学习模块：手写数字识别与垃圾分类

3.1 实验内容

LeNet5 + MNIST 被誉为深度学习领域的 “Hello world”。本实验主要介绍使用 MindSpore 在 MNIST 手写数字数据集上开发和训练一个 LeNet5 模型，并验证模型精度。

通过以上学习，使用 MindSpore 深度学习框架实现 26 种垃圾进行分类。

MindSpore 是最佳匹配 Ascend（昇腾）芯片的开源 AI 计算框架，同时也支持 CPU、GPU 平台。访问 MindSpore 官网了解更多：
<https://www.mindspore.cn/>

深度学习计算中，从头开始训练一个实用的模型通常非常耗时，需要大量计算能力。常用的数据如 OpenImage、ImageNet、VOC、COCO 等公开大型数据集，规模达到几十万甚至超过上百万张。网络和开源社区上通常会提供这些数据集上预训练好的模型。大部分细分领域任务在训练网络模型时，如果不使用预训练模型而从头开始训练网络，不仅耗时，且模型容易陷入局部极小值和过拟合。因此大部分任务都会选择预训练模型，在其上做微调（也称为 Fine-Tune）。

本实验以 MobileNetV2+ 垃圾分类数据集为例，主要介绍如在使用 MindSpore 在 CPU/GPU 平台上进行 Fine-Tune（微调）。

3.2 实验要求

- (1) 了解如何使用 MindSpore 进行简单卷积神经网络的开发。
- (2) 了解如何使用 MindSpore 进行简单图片分类任务的训练。
- (3) 了解如何使用 MindSpore 进行简单图片分类任务的验证。

3.3 预备知识

(1) 熟练使用 Python，了解 Shell 及 Linux 操作系统基本知识。并在 cpu/gpu 下进行网络微调

(2) 具备一定的深度学习理论知识，如卷积神经网络、损失函数、优化器，训练策略等。

(3) 了解并熟悉 MindSpore AI 计算框架，MindSpore 官网：<https://www.mindspore.cn>

(4) 学 习 更 多 MindSpore 实 验 案 例 ， 请 前 往：

<https://gitee.com/mindspore/course>

(5) 数据集下载地址：[https://ascend-professional-construction-](https://ascend-professional-construction-dataset.obs.cn-north-4.myhuaweicloud.com:443/MindStudio-pc/data_en.zip)

[dataset.obs.cn-north-4.myhuaweicloud.com:443/MindStudio-pc/data_en.zip](https://ascend-professional-construction-dataset.obs.cn-north-4.myhuaweicloud.com:443/MindStudio-pc/data_en.zip)

(6) 参考资料：<https://bbs.huaweicloud.com/forum/thread-83710-1-1.html>

3.4 实验环境

MindSpore 深度学习框架

3.5 参考资料

<https://www.mindspore.cn>

4. 大模型：古诗转水墨画

4.1 实验内容

古诗和水墨画都是中华文化的重要组成部分，它们源远流长、博大精深，是中国文化的瑰宝。古诗是中国文学的珍品，它以简洁、含蓄、意境深远的语言表达着人们对自然、人生、情感等方面的感悟和体验，具有较高的艺术价值和文化内涵。水墨画则是中国绘画的典型代表，它以黑白墨线和水墨渲染为主要表现手法，通过形似神似、运笔精妙的艺术手法，来充分表现山水、花鸟、人物等各种形态的艺术形象，具有极高的传承价值。

本实验内容旨在利用 Stable Diffusion 等深度学习技术，实现根据古诗自动生成水墨画的功能。实验参与者将学习 Python 编程语言在深度学习中的应用，并进一步掌握基于 Python 语言的深度学习框架（Pytorch）。通过对古诗进行自然语言处理和图像风格转换的技术实现，参与者将了解可控生成深度学习大模型的基本原理和关键技术，并实际体验模型效果。

4.2 实验要求

- 1) 已提供 Stable Diffusion 大模型的代码与预训练参数
- 2) 已提供训练和测试数据
- 3) 要求参与者自行部署 Stable Diffusion 大模型，并自行训练和调参
- 4) 将训练好的最优模型提交到指定平台进行测试

4.3 选做内容(加分项)

- 1) 训练模型生成写实画作
- 2) 分析模型的实际效果，指出如何改进模型，进一步提升效果
- 3) 自行换用其他可控生成模型，尝试效果

4.4 实验要求

要求使用 Pytorch 深度学习框架完成本次实验，允许使用基于 Python 的 Pandas、Numpy、Sklearn 等算法库进行数据处理。

不同 Python 库需要版本匹配才能正常运行。

4.5 参考资料

Python:

<https://www.runoob.com/python3/python3-tutorial.html>

Numpy:

<https://www.numpy.org>

Sklearn:

<https://scikit-learn.org>

Pytorch:

<https://pytorch.org/>

参考文献:

High-Resolution Image Synthesis with Latent Diffusion Models

提交内容

课程设计报告、完整源代码。

注意：

- ①严禁抄袭，代码、报告将进行查重与人工复查，一经发现任何形式的抄袭（包括但不限于：报告内容大面积重叠、代码核心部分构造相似等），抄袭者与被抄袭者均判为不及格。
- ②报告尽量控制在 30 页以内（不含封面），超过太多将酌情进行扣分。
- ③严格遵守版权，否则成绩将判为不及格。

报告合计 100 分，占比 70%，评分项及细则如下：

评分项	分值	细则
问题概述	15	<p>说明：</p> <p>针对项目需要解决的问题，给出关于问题的描述和对问题的分析。包括但不限于以下内容：</p> <ul style="list-style-type: none">1) 问题的直观描述2) 项目已有代码的阅读和理解3) 解决问题的思路 and 想法 <p>评分细则</p> <ul style="list-style-type: none">1) 清晰且充分地描述与分析（15 分）2) 描述与分析不够清晰或不够充分（5-14 分）3) 未进行问题的描述与分析（0 分）
算法设计	15	<p>说明：</p> <p>描述解决问题所使用的算法的原理，要求用自己的语言来表达。包含但不限于：算法功能、设计思路、算法流程图</p>

		<p>等内容。</p> <p>评分细则：</p> <p>1) 内容完整，原理正确（15 分）</p> <p>2) 内容存在遗漏，算法存在错误（5-14 分）</p> <p>3) 未进行算法设计的说明（0 分）</p>
算法实现	15	<p>说明：</p> <p>在算法原理的基础上，结合代码（以伪代码或者标明注释的源代码）讲述算法的实现细节、核心函数、模块输入输出、数据结构定义等内容。</p> <p>评分细则：</p> <p>1) 内容完整，实现正确（15 分）</p> <p>2) 内容存在遗漏，算法存在错误（5-14 分）</p> <p>3) 未进行算法设计的说明（0 分）</p>
实验结果	40	<p>说明：</p> <p>对实验结果进行详细展示，对于每个问题展示测试截图，对于测试用例进行描述说明，对于未通过测试的用例结合自己的算法进行分析，建议结合调试过程进行说明。</p> <p>评分细则：</p> <p>该评分项将根据项目要求的 Question 完成情况以及报告说明的完整性进行综合评分。其中每个 Question 所占分值已在项目网址中写明，共计 25 分，其余分</p>

		数则根据报告中实验结果分析的完整度、清晰度来进行给分，共计 15 分。
总结与分析	10	<p>说明：</p> <p>分析实验中遇到的问题及解决方案，总结关于本项目的收获与思考。包含但不限于：对算法的理解、优缺点的评价、算法的适用场景等。</p> <p>评分标准：</p> <p>1) 内容完整，且能体现独立思考的过程（10 分）</p> <p>2) 内容较为完整，较能体现独立思考的过程（5-9分）</p> <p>3) 内容残缺，不能体现独立思考的过程（0 分）</p>
格式规范	5	<p>说明：</p> <p>正文五号宋体，行距固定 18 磅。其余字体、字号、段落、插图、代码块等格式需调整至美观。</p> <p>评分标准：</p> <p>1) 结构清晰，格式美观（5 分）</p> <p>2) 结构完整，格式不美观但不影响评阅（1-4 分）</p> <p>3) 结构残缺，格式凌乱且严重影响评阅（0 分）</p>
参考文献	0	<p>说明：</p> <p>若有参考内容，需要标注出来</p>

代码合计 100 分，占比 30%，评分项及细则如下：

代码文件	100	<p>说明：</p> <p>项目代码需提交整个压缩包文件，既检验各个问题的正确性也考量代码的规范性，包含但不限于：代码编写规范、变量定义清晰、注释清晰、代码便于理解和阅读。</p>
------	-----	--