**附件一： 《人工智能导论》课程设计报告参考选题**

**说明：鼓励自主选题**，若自主选题较难，可参考如下选题。

选题1、五子棋游戏的设计与实现

选题2、十五数码问题的求解实现

选题3、汉诺塔动态模拟过程的设计与实现

选题4、智能迷宫求解的设计与实现

选题5、智能旅行商问题（TSP）求解的设计与实现

选题6、智能猴子拿香蕉问题的求解策略与实现

选题7、动物识别方法的设计与实现

选题8、基于A\*算法求解八数码问题

选题9、基于A\*算法的最优路径规划系统

选题10、A\*算法的改进研究

选题11、图搜索算法对比研究

选题12、基于α-β剪枝算法的五子棋游戏

选题13、五子棋机器博弈系统

选题14、基于回溯搜索的地图着色方法设计与实现

选题15、用合适的算法(如遗传算法等)求解TSP问题

选题16、家族人物关系推理系统

选题17、数独游戏

选题18、基于机器学习的垃圾邮件分类系统

选题19、基于深度学习的图像识别应用—车牌识别

选题20、基于自然语言处理(NLP)的智能客服系统

选题21、基于深度学习的图像分类和识别系统设计

选题22、基于知识图谱的医疗诊断应用

选题23、基于深度学习的人脸识别与身份验证系统

选题24、基于深度学习的推荐系统设计

选题25、基于机器学习的社交媒体情感分析

选题26、基于深度学习的五线谱识别及其在构建音乐知识图谱中的应用

**选题1、五子棋游戏的设计与实现**

**1、概述**

通过实现一个五子棋游戏，将人工智能中的搜索算法、评估函数，以及可能的机器学习或深度学习技术，应用于游戏AI的设计中。了解游戏AI的基本原理，掌握如何在复杂游戏环境中应用启发式搜索算法（如Alpha-Beta剪枝、极小极大搜索、迭代加深度优先搜索等），并尝试为五子棋游戏设计一个有效的评估函数。此外，在初步实现情况下，鼓励探索利用神经网络和强化学习技术，进一步提升AI的智能水平。

**2、具体内容**

⑴、五子棋游戏规则与界面设计

①介绍五子棋的基本规则。

②设计并实现一个简单的五子棋游戏界面，支持人机对战或双人对战模式。

⑵、基础搜索算法实现

①实现极小极大搜索算法，用于游戏树的遍历和决策。

②实现Alpha-Beta剪枝算法，以减少搜索空间，提高搜索效率。

③（可选）实现迭代加深度优先搜索，以处理游戏树深度未知的情况。

⑶、评估函数设计

①设计一个有效的评估函数，用于估计棋盘上的局势，指导AI的决策。

②评估函数可以基于棋子的数量、位置、形状等因素进行设计。

⑷、游戏AI设计与实现

①将搜索算法和评估函数结合，实现五子棋游戏AI。

②测试AI在不同难度设置下的表现，调整评估函数和搜索参数以优化AI性能。

⑸、（可选）机器学习/深度学习技术探索：

①（可选）探索使用神经网络（如卷积神经网络）评估棋盘局势，或利用强化学习技术训练五子棋AI。

②收集棋谱数据，训练神经网络模型，并评估其在五子棋游戏中的表现。

⑹、系统测试与评估

①对五子棋游戏系统进行测试，包括功能测试、性能测试和用户体验测试。

②评估AI在不同难度下的表现，记录并分析测试结果。

**选题2、十五数码问题的求解实现**

**1、概述**

通过实现一个十五数码问题的求解器，深入探索人工智能中的搜索算法、状态空间表示、路径规划等关键技术。学习如何表示十五数码问题的状态空间，应用启发式搜索算法（如A\*算法、宽度优先搜索BFS、深度优先搜索DFS等）寻找问题的解，并设计有效的启发式函数来指导搜索过程，从而提高求解效率。此外，还可以探索利用约束满足问题（CSP）求解技术或遗传算法等高级方法来进一步解决或优化该问题。

**2、具体内容**

⑴、十五数码问题规则介绍

①介绍十五数码问题的规则和目标。

②描述问题的状态空间及其表示方法，如使用二维数组或邻接表等数据结构。

⑵、状态空间搜索算法实现

①实现宽度优先搜索（BFS）和深度优先搜索（DFS）算法，并比较其性能和适用性。

②实现A\*算法，设计并实现有效的启发函数，引导搜索过程，减少搜索空间。

⑶、启发式函数设计

①设计并实现一个启发式函数，用于评估当前状态与目标状态之间的距离或差异。

②（可选）尝试设计不同启发式函数，及其对搜索效率和解的影响。

⑷、算法优化与性能提升

①探索并应用剪枝技术，以减少搜索空间和提高搜索效率。

②（可选）实现约束满足问题（CSP）求解技术或遗传算法等高级方法，尝试解决或优化十五数码问题。

⑸、系统测试与评估

①对十五数码问题求解器进行测试，包括功能测试、性能测试。

②记录并分析测试结果，评估求解器的效率和可靠性。

⑹、用户交互界面设计

设计实现一个简单的用户交互界面，允许用户输入初始状态并查看求解过程及结果。

**选题3、汉诺塔动态模拟过程的实现与实现**

**1、概述**

通过设计一个汉诺塔动态模拟系统，深入理解汉诺塔问题的求解原理，掌握递归算法的应用，同时提升编程实践和界面设计能力。要求将一系列大小不同、穿在一根柱子上的圆盘移动到另一根柱子上，移动过程中应满足以下规则：

⑴、每次只能移动一个圆盘。

⑵、任何时候都不能将一个大圆盘放在小圆盘的上面。

⑶、可以借助第三根柱子作为辅助。

⑷、问题的目标是找到将所有圆盘从初始柱子移动到目标柱子的最小移动步骤。

**2、具体内容**

⑴、问题理解：深入理解汉诺塔问题的规则和目标，明确问题的求解步骤。

⑵、算法设计：设计并实现一个递归算法来求解汉诺塔问题，确保算法的正确性和效率。

⑶、界面设计：设计一个用户友好的图形界面，用于展示汉诺塔的动态移动过程。

⑷、系统实现：将算法和界面集成在一起，形成一个完整的汉诺塔动态模拟系统。

**3、扩展功能（可选）**

⑴、动画效果优化：为圆盘的移动添加更丰富的动画效果，如渐变、旋转等，使移动过程更加生动。

⑵、启发式搜索算法实现：尝试使用启发式搜索算法（如A\*算法），求解汉诺塔问题，比较其与递归算法的性能差异。

⑶、多用户支持：尝试支持多个用户同时进行操作，展示不同用户的移动过程。

**选题4、智能迷宫求解的设计与实现**

**1、概述**

走迷宫问题是人工智能中经典的搜索问题之一，它涉及到路径搜索、状态空间表示、启发式搜索算法等多个核心概念。本选题旨在通过设计一个智能迷宫求解系统，深入理解迷宫问题的求解原理，掌握各种搜索算法的应用，同时提升编程实践和问题解决能力。

**2、具体内容**

⑴、问题理解：深入理解迷宫问题的定义、求解目标和约束条件。

⑵、状态空间表示：设计合适的数据结构来表示迷宫的状态空间，包括迷宫地图、当前位置、目标位置等。

⑶、搜索算法设计：设计并实现至少两种搜索算法（如深度优先搜索、宽度优先搜索、A\*搜索等），求解迷宫问题。

⑷、界面设计：设计一个用户友好的图形界面，用于展示迷宫的地图、当前位置、路径等信息，并提供算法选择和参数设置功能。

⑸、系统实现：将算法和界面集成在一起，形成一个完整的智能迷宫求解系统。

**3、扩展功能（可选）**

⑴、启发式函数设计：为A\*搜索算法设计更高效的启发式函数，以提高搜索效率。

⑵、动态迷宫生成：设计算法自动生成不同规模和复杂度的迷宫地图，增加系统的趣味性和挑战性。

⑶、多路径求解：实现算法找到多条从起点到终点的路径，并允许用户选择最优路径或展示所有路径。

⑷、实时路径更新：在用户界面上实时显示搜索过程中的路径变化，增加系统的动态性和可视化效果。

**选题5、智能旅行商问题（TSP）求解的设计与实现**

**1、概述**

旅行商问题（Traveling Salesman Problem, TSP）是人工智能中经典的组合优化问题之一，它涉及到路径规划、图论、启发式搜索算法等多个核心概念。本选题旨在通过设计一个智能旅行商问题求解系统，深入理解TSP问题的求解原理，掌握各种搜索算法和近似算法的应用，同时提升编程实践和问题解决能力。

**2、具体内容**

⑴、问题理解：深入理解TSP问题的定义、求解目标和约束条件。

⑵、图论基础：了解图论的基本概念，包括节点、边、权重等，以及图的表示方法（如邻接矩阵、关联矩阵等）。

⑶、搜索算法设计：设计并实现至少两种搜索算法（如A\*算法、宽度优先搜索算法、深度优先搜索算法等）或近似算法（如最近邻算法、遗传算法、蚁群算法等）求解TSP问题。

⑷、界面设计：设计一个用户友好的图形界面，用于展示城市地图、当前路径、最优路径等信息，并提供算法选择和参数设置功能。

⑸、系统实现：将算法和界面集成，形成一个完整的智能旅行商问题求解系统。

**3、扩展功能（可选）**

⑴、动态城市生成：设计算法自动生成不同规模和分布的城市地图，增加系统的趣味性和挑战性。

⑵、多起点/多终点问题：扩展系统以处理多起点或多终点的TSP问题，或允许用户自定义访问城市的顺序。

⑶、实时路径更新：在用户界面上实时显示搜索过程中的路径变化，增加系统的动态性和可视化效果。

⑷、启发式函数优化：对于启发式搜索算法，设计更高效的启发式函数以提高搜索效率。

⑸、并行与分布式计算：利用多线程或分布式计算技术加速算法的执行，提高系统的处理能力。

**选题6、智能猴子拿香蕉问题的求解策略与实现**

**1、概述**

猴子拿香蕉问题是人工智能中用于介绍搜索算法、问题表示、状态空间搜索等概念的经典案例。该问题通常描述为：一只猴子被关在一个房间里，房间里有一些障碍物，香蕉放在房间的某个位置，猴子需要通过移动来避开障碍物并拿到香蕉。本选题旨在通过实现一个智能猴子拿香蕉问题的求解系统，深入理解搜索算法的原理，掌握状态空间表示、启发式搜索等技术的应用，同时提升编程实践和问题解决能力。

**2、具体内容**

⑴、问题理解：理解猴子拿香蕉问题的目标、约束条件和可能的解决方案。

⑵、状态空间表示：设计一种合适的数据结构来表示猴子、箱子、香蕉的状态空间。

⑶、搜索算法设计：设计并实现至少两种搜索算法（如深度优先搜索、广度优先搜索、A\*搜索等），求解猴子拿香蕉问题。

⑷、启发式函数设计：若采用启发式搜索算法，需设计一个有效的启发式函数来评估状态的好坏，以指导搜索方向。

⑸、界面设计：设计一个用户友好的图形界面，用于展示房间布局、猴子移动、箱子位置、香蕉位置等信息。

⑹、系统实现：将算法和界面集成，形成一个完整的智能猴子拿香蕉问题求解系统。

**选题7、动物识别方法的设计与实现**

**1、概述**

动物识别问题是人工智能中的一个经典应用案例，它涉及到图像识别、特征提取、分类算法等多个方面。本选题旨在通过实现一个动物识别系统，深入理解动物识别的基本原理，掌握产生式系统和机器学习方法在动物识别中的应用，并比较这两种实现方式在性能、可解释性、灵活性等方面的差异，增强编程实践、问题解决以及系统设计与集成能力。

**2、具体内容**

⑴、问题理解：理解动物识别问题的目标、约束条件和可能的解决方案。

⑵、数据集准备：利用可获取的动物图像数据集，训练和测试动物识别模型。

⑶、特征提取：设计算法提取图像中的关键特征，如颜色、纹理、形状等，用于后续的分类任务。

⑷、产生式系统实现

①设计产生式规则：基于动物特征，设计一系列产生式规则，用于识别不同的动物。

②实现推理引擎：编写代码实现推理引擎，根据输入的特征和产生式规则进行推理，输出识别结果。

⑸、机器学习方法实现

①选择合适的机器学习算法：如支持向量机（SVM）、卷积神经网络（CNN）等，用于训练动物识别模型。

②模型训练：使用数据集训练机器学习模型，调整模型参数以提高识别准确率。

③模型测试：使用测试数据集评估模型的性能，包括准确率、召回率等指标。

⑹、性能比较：比较产生式系统和机器学习方法在动物识别任务中的性能差异，包括识别准确率、计算效率、可解释性等方面。

⑺、系统实现（可选）：将产生式系统和机器学习方法集成在一起，形成一个完整的动物识别系统，并提供用户友好的界面进行交互。

**选题8、基于A\*算法求解八数码问题**

⑴、至少定义3种不同的启发式函数，编程实现求解八数码问题的A\*算法；

⑵、要求用可视化界面演示算法执行过程，应能选择预定义的启发式函数，能随机初始化初始状态，能单步执行，也能连续执行，能画出搜索树，同时标出估价函数在每个节点的各项函数值，能展示OPEN表和CLOSED表的动态变化过程；

⑶、能统计出扩展节点数和算法执行时间，以便对采用不同启发式函数的A\*算法的性能做对比研究。

**选题9、基于A\*算法的最优路径规划系统**

⑴、基于真实地图实现，可以是位图背景加栅格坐标数据，也可以直接使用某种格式的GIS（地理信息系统）矢量地图，地图规模不能太小；

⑵、可以设置起点和终点；

⑶、要求用可视化界面演示算法执行过程，能单步执行，也能连续执行，画出扩展过的所有路径，画出最优路径，能展示OPEN表和CLOSED表的动态变化过程；

⑷、可考虑路况信息，改进启发式函数，以求更实用。

**选题10、A\*算法的改进研究**

⑴、给出改进思路并编程实现改进的算法；

⑵、结合一个具体问题实验对比改进前后的算法性能。

**选题11、图搜索算法对比研究**

⑴、结合某个具体例子，编程实现宽度优先、等代价、深度优先、有序搜索算法；

⑵、要求用可视化界面演示算法执行过程，能单步执行，也能连续执行，能画出搜索树，能展示OPEN表和CLOSED表的动态变化过程；

⑶、可以自定义搜索图，通过实验研究各种图搜索算法的性能。

**选题12、基于*α*-*β*剪枝算法的五子棋游戏**

⑴、编写五子棋游戏程序，支持人机对战；

⑵、编程实现*α*-*β*剪枝算法，作为机器方的下棋算法。

**选题13、五子棋机器博弈系统**

⑴、编程实现一个五子棋主控程序，要求有可视化棋盘，有裁判功能，支持通过Socket接口连接选手，有清晰简洁的通信协议，支持循环赛赛程管理；

⑵、编写一个五子棋下棋算法，通过Socket接口接入主控程序，与其他机器选手对战。

**选题14、基于回溯搜索的地图着色方法**

⑴、对中国地图中的省级行政区进行着色，最多使用四种颜色；

⑵、编程实现回溯算法用于地图自动着色；

⑶、研究回溯的改进算法，并编程实现。

**选题15、用合适的算法(如遗传算法等)求解TSP问题**

⑴、编程实现遗传算法，允许自定义算法参数；

⑵、要求用可视化界面演示算法执行过程，设置一个演示速度参数，以使演示进度可控；

⑶、提供自定义TSP问题的接口。

**选题16、家族人物关系推理系统**

⑴、简单构建《红楼梦》中以贾家为中心的人物关系图谱；

⑵、任给图中两个人物，能推理出二者之间的关系；

⑶、（可选）用可视化界面演示推理过程。

**选题17、数独游戏**

⑴、设计数独游戏，能自动生成初盘，也能人工设置初盘，能检测人工设置初盘的合法性；

⑵、编写一个求解数独终盘的算法。

**选题18、基于机器学习的垃圾邮件分类系统**

⑴、构建一个能够自动识别并分类垃圾邮件的系统，提高个人或企业邮箱的使用效率。

⑵、收集并预处理邮件数据集，包括提取邮件主题、正文、发件人等特征。

使用Python和scikit-learn库实现一个基于朴素贝叶斯、支持向量机或随机森林等算法的垃圾邮件分类器。

⑶、划分训练集和测试集，训练模型并评估其性能，如准确率、召回率等。

⑷、设计用户界面（可选），使系统能够在实际邮箱环境中运行，自动分类邮件。

⑸、实现一个可在实际邮箱环境中运行的垃圾邮件分类系统，提高邮件处理的效率和准确性。

**选题19、基于深度学习的图像识别应用—车牌识别**

⑴、开发一个车牌识别系统，用于停车场管理、交通监控等场景。

⑵、收集并预处理车牌图像数据集，包括图像增强、裁剪、标注等。

⑶、使用TensorFlow、PyTorch或Mindspore框架，实现一个卷积神经网络（CNN）模型，用于车牌字符识别。

⑷、训练模型，并调整网络架构、优化器及学习率等参数以提高识别准确率。

⑸、设计一个用户界面（可选），使系统可在实际场景中运行，识别并显示车牌信息。

⑹、实现一个能够在实际场景中准确识别车牌字符的系统，为停车场管理、交通监控等提供便利。

**选题20、基于自然语言处理(NLP)的智能客服系统**

⑴、开发一个可自动回答用户问题、提供帮助的智能客服系统，用于电商、金融等领域。

⑵、收集并预处理用户问题数据集，包括问题分类、关键词提取等。

⑶、使用Python和NLTK、SpaCy等自然语言处理库，实现一个基于规则或机器学习的问答系统。

⑷、训练模型，使其能够准确理解用户问题并给出相应回答。

⑸、设计用户界面（如网页、APP等），使系统能够在实际客服环境中运行。

⑹、实现一个能够在实际客服环境中，自动回答用户问题的智能客服系统，提高客服效率和用户满意度。

**选题21、基于深度学习的图像分类和识别系统设计**

**1、理论学习**

⑴、学习并掌握人工智能、机器学习和深度学习的基础理论，特别是卷积神经网络（CNN）在图像分类和识别中的应用。

⑵、研究AlexNet、VGG、ResNet等经典CNN模型的结构、特点和性能，理解它们如何在图像分类任务中取得显著成果。

**2、系统设计**

⑴、设计一个基于深度学习的图像分类和识别系统的整体架构，包括数据输入、预处理、模型训练、预测和评估等模块。

⑵、设计一个系统，用于识别猫狗图像。系统架构包括图像输入模块（接收用户上传的图像）、预处理模块（对图像进行缩放、裁剪和归一化等操作）、模型训练模块（使用CNN模型进行训练）、预测模块（对输入图像进行分类预测）和评估模块（计算分类准确率等评估指标）。

**3、数据准备：**

⑴、收集并整理图像数据集，确保数据的质量和多样性，并进行预处理。

⑵、使用公开的猫狗图像数据集（如Kaggle上的Dogs vs. Cats数据集），对数据进行清洗（去除模糊、重复图像）、增强（如旋转、翻转、调整亮度等）和归一化处理，以提高模型的泛化能力。

**4、模型训练与优化**

⑴、选择或设计深度学习模型，进行模型训练，并调整超参数以优化模型性能。

⑵、选择ResNet模型作为分类器，使用预处理后的猫狗图像数据集进行训练。通过调整学习率、批量大小、迭代次数等超参数，以及使用正则化、dropout等技术防止过拟合，提高模型的分类准确率。

**5、系统实现与测试**

⑴、编写代码实现系统的各个模块，并进行功能测试和性能测试。

⑵、使用Python和TensorFlow/PyTorch/Mindspore等深度学习框架编写代码，实现猫狗图像分类系统。通过测试集对系统进行测试，计算分类准确率、召回率等评估指标，确保系统的稳定性和准确性。

**选题22、基于知识图谱的医疗诊断应用**

**1、理论学习**

⑴、学习人工智能、机器学习、深度学习以及知识图谱技术的基础理论，理解这些技术在医疗诊断中的应用原理和优势。

⑵、学习医疗诊断的基础知识，包括常见疾病的诊断方法、医学影像的解读等，以便更好地理解人工智能在医疗诊断中的实际作用。

**2、知识图谱构建**

⑴、研究医疗领域的知识图谱构建方法，包括数据收集、实体识别、关系抽取、图谱存储等。

⑵、构建医疗领域的知识图谱，涵盖疾病、症状、药物、治疗等实体及其之间的关系，为医疗诊断提供丰富的知识支持。

**3、算法设计与实现**

⑴、设计基于知识图谱的医疗诊断算法，利用图谱中的实体和关系进行推理和分析，辅助医生进行诊断。

⑵、实现算法的核心功能，包括知识检索、路径推理、语义匹配等，确保算法能够准确、高效地处理医疗诊断任务。

**4、系统设计与开发**

⑴、设计一个基于知识图谱的医疗诊断系统，包括系统的整体架构、功能模块、数据流程等。

⑵、开发系统的关键功能模块，如用户交互界面、知识图谱管理、诊断推理引擎等，确保系统能够稳定运行并提供良好的用户体验。

**5、实验与评估**

⑴、使用真实的医疗数据集对系统进行测试，评估系统的诊断性能和准确性。

⑵、分析系统在处理不同疾病、不同症状等方面的表现，找出系统的优点和不足。

⑶、根据测试结果，对系统进行优化和改进，提高系统的实用性和可靠性。

**选题23、基于深度学习的人脸识别与身份验证系统**

**1、理论学习**

⑴、学习深度学习的基础理论，包括卷积神经网络（CNN）、损失函数、优化算法等。

⑵、了解人脸识别和身份验证的基本原理，以及深度学习在这些领域的应用。

**2、数据集准备**

⑴、收集并标注人脸图像数据集，确保数据集的质量和多样性。

⑵、对数据集进行预处理，如调整图像大小、归一化等，以适应深度学习模型的输入要求。

**3、模型选择与训练**

⑴、选择适合人脸检测和识别的深度学习模型，如MTCNN用于人脸检测，FaceNet或DeepFace用于人脸识别。

⑵、使用预处理后的数据集对模型进行训练，调整模型参数以优化检测和识别效果。

**4、系统开发**

⑴、设计并实现后端服务，包括模型训练、API开发等，为前端提供接口支持。

⑵、开发前端界面，实现用户界面和交互设计，包括人脸图像上传、识别结果显示等功能。

**5、系统测试与优化**

⑴、进行单元测试、集成测试和性能测试，确保系统的稳定性和准确性。

⑵、根据测试结果对系统进行优化和改进，提高系统的实用性和可靠性。

**选题24、基于深度学习的推荐系统设计**

**1、理论学习**

⑴、学习深度学习的基础理论，包括但不限于卷积神经网络（CNN）、循环神经网络（RNN）、长短期记忆网络（LSTM）、自注意力机制、图神经网络（GNN）等。

⑵、了解推荐系统的基本原理，包括协同过滤、基于内容的推荐、混合推荐等。

⑶、掌握深度学习在推荐系统中的应用，如神经协同过滤、深度矩阵分解等。

**2、数据集准备**

⑴、收集并标注推荐系统所需的数据集(如FilmTrust、Epinions、Flixster、Douban等)，这些数据集应包含用户行为数据、物品属性数据等。

⑵、对数据集进行预处理，如数据清洗、去重、归一化等，确保数据质量和一致性。

**3、模型选择与训练**

⑴、根据推荐系统的需求和数据特点，选择合适的深度学习模型。

⑵、使用预处理后的数据集对模型进行训练，调整模型参数以优化推荐效果。

⑶、训练过程中，可以使用反向传播算法和优化器来更新模型参数。

**4、系统开发**

⑴、设计并实现推荐系统的后端服务，包括模型训练、预测、数据存储等。

⑵、开发前端界面，实现用户界面和交互设计，包括推荐结果展示、用户反馈等功能。

⑶、确保系统的可扩展性和实时性，以满足实际应用场景的需求。

**5、系统测试与优化**

⑴、进行单元测试、集成测试和性能测试，确保系统的稳定性和准确性。

⑵、根据测试结果对系统进行优化和改进，提高推荐效果和用户体验。

**选题25、基于机器学习的社交媒体情感分析**

**目标：**通过机器学习技术，分析社交媒体上的文本数据，以识别和理解用户的情感倾向。

**1、数据收集与预处理**

⑴、从社交媒体平台（如微博等）收集包含情感色彩的文本数据。

⑵、确保数据的代表性、多样性和准确性，为后续分析提供可靠的基础。

⑶、使用微博官方的API对微博进行抓取，分类标注后作为情感分析的数据集。

**2、特征提取**

⑴、从文本数据中提取与情感倾向相关的特征，如词频、词袋模型、TF-IDF、词嵌入等。

⑵、将文本数据转化为数值向量，以便机器学习算法能够处理和分析。

⑶、使用词袋模型将每个文本表示为一个向量，其中向量的每个维度表示一个单词在文本中出现的次数。

**3、模型选择与训练**

⑴、选择合适的机器学习算法（如支持向量机、朴素贝叶斯分类器、深度学习模型等）进行情感分析。

⑵、通过训练模型来自动分析文本数据的情感倾向，如积极、消极或中性。

⑶、使用朴素贝叶斯算法对微博进行情感分析，将微博分为积极、消极、客观三类，并使用AdaBoost算法对朴素贝叶斯算法进行加强。

**4、模型评估与优化**

⑴、使用交叉验证、准确率、召回率、F1分数等指标评估模型的性能，并根据评估结果进行模型优化。

⑵、提高模型的准确性和泛化能力，确保情感分析的可靠性。

⑶、使用GridSearchCV类搜索最优的超参数，并使用交叉验证来评估模型的性能。

**5、结果可视化与应用**

⑴、将情感分析的结果进行可视化展示，如情感分布图、情感趋势图等。同时，将情感分析应用于实际场景，如品牌声誉监测、舆情分析等。

⑵、直观展示情感分析结果，为决策制定提供支持；通过实际应用验证情感分析的有效性和实用性。

**选题26、基于深度学习的五线谱识别及其在构建音乐知识图谱中的应用**

本选题旨在探索深度学习技术在音乐信息处理领域的创新应用，通过实现五线谱的小节分割与音乐符号的精确识别，进而构建音乐知识图谱，为音乐学习、创作、分析及信息检索等提供智能化支持。该研究不仅能促进人工智能与音乐学科的交叉融合，还可深化对音乐信息结构化表示的理解。

**1、内容概述**

探索深度学习技术在音乐信息处理领域的应用，针对五线谱图像的分析与理解，实现从五线谱图像中分割出独立的小节，然后识别每个小节内的各种音乐符号（如音符、位置、休止符、动态标记、调号等），最终利用这些识别结果构建音乐知识图谱。该知识图谱能够结构化地表示乐曲中的音乐元素及其关系，为音乐分析、教学辅助、作曲自动化等应用提供支持。

**2、主要任务**

⑴、五线谱小节分割算法设计与实现

研究并设计一种高效的图像分割算法，能够准确地将五线谱图像分割成包含单个或多个小节的子图像。考虑使用基于卷积神经网络（CNN）的语义分割技术，结合五线谱特有的结构特征进行优化。

⑵、音乐符号识别模型的设计与实现

针对分割后的小节图片，构建并训练深度学习模型（如深度卷积神经网络、循环神经网络或两者的组合），识别其中的各类音乐符号。过程中需收集并标注高质量的五线谱图像数据集，涵盖多种音乐符号类型。

⑶、音乐知识图谱的构建

①定义音乐知识图谱的节点和边类型，如音符节点、小节节点、音符间的时序关系、和弦关系等。

②利用识别出的音乐符号信息，将它们映射到知识图谱的相应节点和边上，形成结构化表示。

③实现知识图谱的存储、查询和可视化功能，便于后续的音乐分析和应用。

⑷、模型的性能评估与优化

①通过实验，对比测试集上的识别准确率、召回率、F1分数等指标，评估小节分割和音乐符号识别的性能。

②根据评估结果，调整模型结构、超参数或引入新的数据增强技术，提高模型整体性能。

**3、目标**

⑴、技术创新：提出一种有效的五线谱小节分割算法和音乐符号识别模型。

⑵、实际应用：构建一个实用的音乐知识图谱系统，能够辅助音乐学习者理解乐谱，为音乐创作提供灵感来源，或为音乐信息检索、智能作曲等高级应用奠定基础。

⑶、理论贡献：探索深度学习在音乐信息处理中的新应用，丰富音乐信息学、人工智能交叉领域的研究，为相关领域的进一步发展提供理论支持和实践经验。

**4、预期成果**

⑴、初步完成一篇包含研究思想、实现方法、实验结果和结论的设计报告。

⑵、（可选）开发一个包含五线谱小节分割、音乐符号识别和构建音乐知识图谱的原型系统。