人工智能辅助编程语言是指专门适应于人工智能和知识工程领域、具有符号处理和 逻辑推理能力的计算机程序设计语言。这类语言能够用来编写程序,求解非数值计算、 知识处理、推理、规划、决策等具有智能的各种复杂问题。

在人工智能领域,选择合适的编程语言非常重要,因为不同的语言在处理特定任务时有其独特的优势。人工智能辅助编程语言是指那些专门为开发人工智能应用而设计的编程语言或框架,它们通常具有高度的抽象性和强大的库支持,能够简化复杂的 AI 算法实现过程。这些语言和框架在 AI 领域有着广泛的应用。编程语言及其特点如表 4-1 所示。

表 4-1 编程语言及其特点

编程语言	特点
Python 语言	易学易用:语法简洁,入门门槛低。 丰富的库:拥有大量的第三方库,极大地简化了机器学习和深度学习的开发工作。 社区支持:活跃的开发者社区,有大量的教程和资源可供学习。 应用场景:机器学习、深度学习、自然语言处理、计算机视觉等。
R语言	统计分析:专为统计计算和图形生成设计,非常适合数据分析和统计建模。强大的可视化:内置丰富的图形库。 学术界广泛使用:在学术界和研究机构中非常流行。 应用场景:数据科学、统计分析、生物信息学等。
Java 语言	跨平台:一次编写,到处运行,适合企业级应用。 性能稳定: Java 虚拟机 (JVM) 提供了良好的性能和稳定性。 丰富的库:有许多成熟的机器学习库。 应用场景:企业级应用、大数据处理、分布式系统等。
GO 语言	并发支持:对于需要大量并行处理的任务,深度学习模型、处理大规模数据集等非常有用。 高性能:编译型特性使其在运行时性能上接近 C/C++,适合高性能计算的 AI 应用。 简洁的语法:语法简洁明了,易于学习和维护,适合快速开发和迭代。 丰富的标准库:Go语言拥有丰富的标准库。

续表

编程语言	特点
Julia 语言	高性能:设计之初就注重性能,执行速度接近 C 语言。 动态类型:语法简洁,易于编写和调试。 科学计算:专为科学计算和数值分析设计,有丰富的数学库。 应用场景:科学计算、数值分析、高性能计算等。
MATLAB语言	矩阵运算:专为矩阵运算设计,非常适合数值计算和线性代数。 可视化:内置强大的图形功能,可以轻松生成各种图表。 工具箱丰富:提供了大量的工具箱,涵盖信号处理、图像处理、控制系统等多个 领域。 应用场景:工程计算、信号处理、图像处理等。

初学者:建议从 Python 开始,因为它简单易学,且有大量的资源和社区支持。

数据科学家: R 是一个不错的选择,特别是如果你主要进行统计分析和数据可视化。

企业级应用: Java 是一个可靠的选择,适合大型项目和企业级应用。

高性能计算: C++和 Julia 适合对性能要求极高的场景。

科学计算: MATLAB 和 Julia 都是不错的选择,特别是如果你需要进行大量的数值计算。

一、AI技术助力编程语言的学习与应用

(一) 人工智能编程语言概述

人工智能编程语言是专门用于开发人工智能应用的编程语言,具有高效、灵活和易 扩展的特点。常见的 AI 编程语言包括 Python、R、Java、C++、GO、Julia、MAT-LAB 等。每种语言都有其独特的优势和应用场景(见图 4-1)。

(二) Python 在 AI 中的应用

Python 因其简洁的语法和丰富的库支持,成为 AI 领域最流行的编程语言之一。常 用的 AI 库包括 NumPy、Pandas、Matplotlib、TensorFlow 和 PyTorch 等。这些库为数 据处理、机器学习和深度学习提供了强大的支持。

(三) 编程语言的选择与学习路径

选择合适的编程语言是学习 AI 的第一步。初学者可以从 Python 开始,逐步掌握基 本语法和常用库的使用。随着学习的深入,可以进一步学习其他编程语言,如 R (用于 统计分析)、Java (用于企业级应用)和 C++ (用于高性能计算)。

(四)案例应用背景

在职业院校教育中,编程语言的学习和应用是一个重要的教学内容。然而,传统的

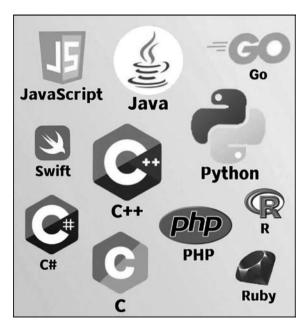


图 4-1 人工智能编程语言

编程教学往往枯燥乏味,学生难以理解和掌握。通过引入 AI 技术,可以使编程学习变得更加生动有趣,激发学生的学习兴趣和创造力。本文将通过一个具体的应用案例,展示如何利用 AI 技术助力编程语言的学习与应用。



AI 技能提升

使用 AI 辅助设计 JavaScript 程序制作乘法口诀表

案例描述:本案例将展示如何使用人工智能 (AI) 辅助设计一个 JavaScript 程序,该程序能够生成并打印出标准的 9×9 乘法口诀表。我们将从基本的 for 循环语句开始,逐步通过 AI 的指导完成整个程序的设计和实现 (见图 4-2)。

for 循环语句是 JavaScript 中一种常用的控制结构,用于反复执行代码块,直到特定条件为 false。以下是 for 循环语句的详细介绍:

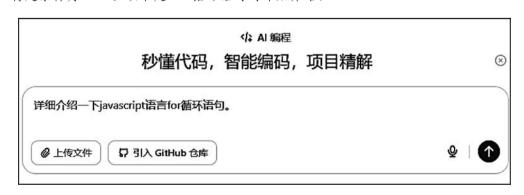


图 4-2 介绍 javascript 语言 for 循环语句

在 JavaScript 中, for 循环是一种常用的迭代语句, 用于重复执行一段代码, 直到满足特定的条件。以下是 for 循环的详细介绍:

1. 基本语法, 如图 4-3 所示。

图 4-3 基本语法

- (1) 初始化表达式:
- ◇通常用于声明并初始化一个或多个计数器变量。
- ◇这个表达式在循环开始前仅执行一次。
- \diamondsuit 例如: let i = 0;
- (2) 条件表达式:
- ◇该表达式在每次循环迭代之前被计算。
- ◇ 如果条件为 true,则执行循环体;如果为 false,则终止循环。
- ◇ 例如: i < 10;。
- (3) 更新语句:
- ◇ 该表达式在每次循环体执行完毕后执行,用于更新计数器变量的值。
- ◇ 例如: i++;。

示例代码:见图4-4。

```
javascript

for (let i = 0; i < 5; i++) {
    console.log(i);
}</pre>
```

图 4-4 示例代码

2. 使用场景

- (1) 遍历数组,正向遍历和反向遍历数组是最常见的操作之一。
- (2) 嵌套循环: 嵌套 for 循环用于处理多维数组或复杂的数据结构。
- (3) 操作对象: 虽然 for 循环通常用于数组, 但也可以用于对象, 通过对象的 键来遍历其属性。

3. 性能优化

(1) 减少数组长度计算:在循环中多次调用 arr. length 会影响性能,可以将长度存储在变量中。

(2) for···of 和 for···in 是更现代的迭代方式,适用于数组、字符串、Map、Set 等可迭代对象。

4. 错误处理和调试

- (1) 无限循环: 确保条件表达式能够在某个时刻变为 false。
- (2) 索引越界: 使用适当的比较运算符。
- (3) 类型错误:确保更新语句能正确改变循环控制变量。



AI 技能提升

人工智能辅助设计 9×9 乘法口诀表

步骤1:理解需求。

首先,我们需要明确乘法口诀表的基本要求。乘法口诀表是一个9行9列的表格,其中第1行第1列的元素是1×1的结果。

步骤 2: 设计算法。

- (1) 使用两个嵌套的 for 循环来遍历乘法表的每一行和每一列。
- (2) 外层循环变量 i 代表行号, 从 1 到 9。
- (3) 内层循环变量;代表列号,也从1到9。
- (4) 在每次内层循环中计算 i×j 的值,并打印出来。
- (5) 每完成一行的打印后,输出一个换行符以分隔不同的行。

步骤 3:编写代码。

根据设计的算法, 我们可以开始编写 JavaScript 代码:

```
}
   </style>
</head>
<body>
   <div id="table-container"></div>
   <script>
       function createMultiplicationTable() {
           const container = document.getElementById('table-container');
           let html = '  ';
           for (let i = 1; i \le 9; i++) {
               html += '';
               for (let j = 1; j <= i; j++) {
                   html += \times \ {i} \times \{i} \times \{i \times j} \left \( td > '; \)
               html += '';
           }
           html += '';
           container. innerHTML = html;
       }
       // 调用函数生成表格
       createMultiplicationTable();
   </script>
</body>
</html>
```

步骤 4: 测试和验证。

运行上述代码,检查控制台输出是否为正确的 9×9 乘法口诀表。确保所有乘积都正确无误,并且格式整洁。

步骤 5: 优化和改进。

根据需要,可以对代码进行优化,例如添加错误处理、用户交互界面等,以提高程序的健壮性和用户体验。

知识拓展

1. 背景

在数字媒体相关专业中,学生经常需要处理图像。图像分类是图像识别领域的

基础任务,对于图像管理和检索有着重要意义。

使用的编程语言与工具:主要使用 Python 语言,借助深度学习框架 Tensor-Flow 和 Keras。Python 简单易学,TensorFlow 和 Keras 为构建图像分类模型提供了便捷的接口(见图 4-5)。



图 4-5 图像分类

2. 实现过程

首先,收集和整理不同类别的图像数据集,如将动物图像分为猫、狗、鸟等类别。

利用 Python 的图像处理库(如 OpenCV 或 Pillow)对图像进行预处理,包括调整图像大小、归一化等操作,使图像数据符合模型输入要求。

使用 TensorFlow 或 Keras 搭建一个简单的卷积神经网络 (CNN) 模型。CNN 在图像分类任务中表现出色,对于职业院校的学生来说,通过简单的模型结构就能初步理解图像特征提取和分类的原理。

对模型进行训练,将预处理后的图像数据输入模型,通过不断调整模型参数, 使其能够准确地识别不同类别的图像。

训练完成后,使用测试数据集对模型进行评估,查看模型的分类准确率等指标。

学习收获:学生通过这个案例,能够学习到图像数据处理的基本方法,了解深度学习模型的搭建和训练过程,掌握利用 Python 实现简单图像分类任务的技能,为今后从事图像相关的工作打下基础。



1. 背景

在信息时代,聊天机器人在客户服务、智能助手等领域应用广泛。对于职业院校计算机相关专业的学生,开发一个简单的聊天机器人可以帮助他们理解自然语言 处理的基本概念和技术。

使用的编程语言与工具:采用 Python 语言,结合自然语言处理库 NLTK (Natural Language Toolkit)和 ChatterBot 库。NLTK 提供了丰富的自然语言处理工具和语料库,ChatterBot 则简化了聊天机器人的开发过程。

2. 实现过程

安装并导入 NLTK 和 ChatterBot 库。

使用 NLTK 对自然语言文本进行预处理,如分词、词性标注等,将文本转化为 计算机能够理解的形式。

创建一个 ChatterBot 实例,并通过训练使其能够理解和回答常见的问题,既可以使用预先准备好的对话数据集进行训练,也可以让机器人在与用户的交互过程中不断学习。

将聊天机器人集成到一个简单的用户界面中,如基于命令行的交互界面或者使用 Tkinter 库创建的图形界面,方便用户与机器人进行对话。

学习收获:通过这个案例,学生可以了解自然语言处理的基本流程,掌握如何使用 Python 库进行文本处理和聊天机器人开发,培养自己在人工智能自然语言处理领域的实践能力,为将来从事相关工作积累经验。

知识拓展

1. 背景

在教育领域,预测学生成绩可以提前发现学习困难的学生并给予帮助。对于职业院校的学生,这是一个具有实际应用价值的案例。

使用的编程语言与工具:运用 Python 语言,结合数据分析库 Pandas、NumPy和机器学习库 Scikit-learn。Pandas 和 NumPy 用于数据处理和分析,Scikit-learn 提供了丰富的机器学习算法。

2. 实现过程

收集学生的学习数据,包括平时作业成绩、考试成绩、出勤情况等。

使用 Pandas 对数据进行清洗和预处理,处理缺失值、异常值等问题,并将数据转化为适合机器学习模型输入的格式。

选择合适的机器学习算法,如线性回归、决策树等,使用 Scikit-learn 构建预测模型。

将数据集划分为训练集和测试集,使用训练集对模型进行训练,调整模型参数 以提高模型的准确性。

使用测试集对训练好的模型进行评估,查看模型的预测性能,如均方误差、决定系数等指标。

学习收获: 学生通过这个案例,可以学习到数据处理和分析的方法,掌握机器学习模型的构建和评估过程,了解如何利用人工智能技术解决教育领域的实际问题,提升自己在数据分析和机器学习方面的能力。



AI 技能提升

智能垃圾分类系统

1. 项目目标

本项目旨在设计并实现一个基于 AI 技术的智能垃圾分类系统。通过该系统,学生不仅可以学习编程语言(如 Python),还能掌握传感器技术、数据处理和机器学习等 AI 相关知识。

2. 项目步骤

(1) 需求分析与系统设计。

需求分析:首先,学生需要明确系统的功能需求,包括垃圾检测、分类、回收等。

系统设计:设计系统的整体架构,包括硬件部分(传感器、控制器等)和软件部分(数据处理、机器学习模型等)。

(2) 硬件搭建。

传感器选择:选择适合的传感器(如摄像头、红外传感器等)来检测垃圾的种 类和位置。

电路设计:设计电路连接传感器和控制模块,确保系统的稳定运行。

(3) 编程实现。

编程语言选择:选择 Python 作为主要的编程语言,因其丰富的库和易学易用的特点。

数据处理:编写代码读取传感器数据,进行数据清洗和预处理。

机器学习模型训练:利用 TensorFlow 或 PyTorch 等框架,训练一个分类模型,用于识别不同种类的垃圾。

系统集成:将数据处理和机器学习模型集成到系统中,实现自动垃圾分类的功能。

(4) 系统测试与优化。

功能测试:在实际环境中测试系统的各项功能,确保其准确性和稳定性。

性能优化:根据测试结果,优化系统的性能,如提高识别准确率、减少响应时间等。

(5) 成果展示。

项目报告: 撰写详细的项目报告, 记录项目的各个环节和实现过程。

成果展示: 在学校的科技节或其他活动中展示项目成果, 接受老师和同学的评价。

3. 操作示范

```
以下是使用 Python 和 TensorFlow 实现垃圾分类系统的一个简单示例:
import tensorflow as tf
import numpy as np
from PIL import Image
# 加载预训练的模型
model = tf.keras.models.load model('garbage classifier.h5')
# 定义图像预处理函数
def preprocess _ image(image _ path):
   img = Image. open(image _ path)
   img = img. resize((128, 128))
   img = np. array(img) / 255.0
   img = np. expand _ dims(img, axis=0)
   return ima
# 测试图像路径
test _ image _ path = 'test _ image. jpg'
# 预处理测试图像
test _ image = preprocess _ image(test _ image _ path)
# 进行预测
prediction = model.predict(test image)
predicted_class = np.argmax(prediction, axis=1)
# 输出预测结果
print(f'Predicted class: {predicted class}')
```

二、人工智能对话及应用

(一) 人工智能对话

1. 人工智能对话基础原则

人工智能对话基础原则是信息转化,即将信息的一种形式转化为另外一种形式,包括以下含义:

语音识别:拥有"思考机制"的进行识别。

信息精简:将一些较长的文章或段落进行摘要,或者是按照要求进行数据分析。

信息创作,按要求创作,依赖对话信息创作文章、编程代码。

2. AI 提示词 (Prompt)

AI 提示词是指用于与人工智能交互时所提供的输入指令或关键词句,用于指定 AI 应该执行的任务和生成的输出。提示词是用户与 AI 之间沟通的桥梁,能够让 AI 理解并据此做出反应或生成创造性内容。

3. 提示词的分类

通用提示词:适用于各种场景的提示词,如"请简要描述""请分析"等。

专业提示词:针对特定领域或行业的提示词,如"请从以下几个方面分析:市场前景、竞争态势、盈利模式等"。

个性化提示词:依照使用者需求定制的提示词,如"请依据以下请求生成一份活动总结报告:字数 1000 字,主题为'人工智能在教育教学领域的应用'"。

4. AI 使用规则

清晰表达——指令明确,目的明确,言简意赅。

对话聚焦——GPT 的对话具有连贯性,所以你单次聊天要聚焦一个主题,这样会更加便捷和精准的得到你想要的答案,如果想开辟一个新话题或主题,新建一个对话开启对话。

重点突出──目的表述要有重点并加深。举例:我想买一台手机→我想买一台玩游戏不卡顿的手机(你要说明你要什么)。

用词专业——如果你聊的话题牵扯到专业领域,可以用一些相关的术语或技术词汇,可以提高问题的准确性和专业性,从而可以获得更具专业性的回答。

内容相关——比如你想有主题的跟 GPT 对话,那就要把你想要的告诉他,比如你想要讨论的是健康/有趣/丰富/愉快等的大学生活,比如学习、生活、饮食等。

5. AI 使用的四大要素——角色设定十背景信息十任务目标十输出要求

设定身份角色、背景信息和任务目标的原因:

功能的局限性: AI 不会自动了解你的内心想法。

产出质量相关:输出质量取决于你的输入信息质量。

个性化服务:根据你的真实意图和输入进行定制化回答。



AI 技能提升

"新小职"带你一起写文案

1. 身份角色:设定 AI 身份十使用者角色。

举例: 你是我的健身教练,负责帮我设计一份每日训练计划,我是一名计算机学院的大一学生。

2. 背景信息:提供上下文信息十原因。

举例: 我是一名职业院校的学生,主修计算机科学。我的目标是在未来一年内掌握机器学习的基本知识和实践技能,以便能够参加校内外的相关竞赛,并为未来的实习机会做好准备。为了实现这一目标,我需要制定一个详细的学习计划。请帮我列出一个涵盖理论学习和实践项目的计划,并推荐一些学习资源。

3. 目标要求:输出什么格式,输出重点要求等。

举例: 我会给你一份本学期我的课程表,请根据我的课程表,生成一份本学期的学习计划,学习计划要包括每周学习任务、体育锻炼安排、学习进度安排、自我测评,可能遇到的问题及详细的解决方案。

(二) AI 应用实践: 撰写校园活动策划书

"新小职"带领同学们利用 AI 技术撰写"趣味运动会"校园活动策划书。包括确定策划书主题和目标、撰写活动方案几个任务点:

策划书是活动的核心内容,使用 AI 辅助设计活动的主题和目标。

使用 AI 工具(如 Kimi)帮助梳理思路,确保主题明确、目标具体。

步骤一:打开"浏览器",输入https://kimi.moonshot.cn。

步骤二: 进入"Kimi" AI 界面, 开启对话。

步骤三:输入:"我是新乡职业技术学院的一名学生,姓名:新小职"(见图 4-6)。

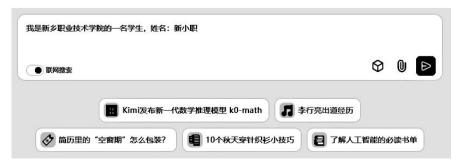


图 4-6 使用 AI 工具 (Kimi) 帮助梳理思路

步骤四: "Kimi"进行人机互动(见图 4-7)。

开启新会话 好,新小职! 很高兴认识你。作为新乡职业技术学院的学生,如果你有任何 问题或者需要帮助,无论是学习上的、生活上的还是其他任何方面,都可以随 时告诉我。我会尽我所能为你提供支持和帮助。有什么可以为你做的吗?

□ 复制 ② 再试一次 「竹分享

图 4-7 "Kimi"进行人机互动

步骤五:输入"我想带领同学们,开展一场趣味运动会,需要设计一份校园活动策划书。请帮我确定策划书主题和目标。"如图 4-8、图 4-9 所示。

当然可以,策划—场趣味运动会是一个很好的团队建设活动。以下是一些建议,帮助你确定策划书的主题和目标:

策算主件收部:

- 1. "活力校园,运动无限" 强调运动的乐趣和校园生活的活力。
- 2. "团结协作,趣味竞技" 突出团队合作和趣味竞技的重要性。
- 3. "健康生活,快乐运动" 以健康和快乐为核心,鼓励学生积极参与运动。
- 4. "挑战自我,超越极限"-鼓励学生挑战自我,体验运动中的成就感。
- 5. "校园奥运,激情飞扬"-模仿奥运会的形式,增加活动的正式感和趣味性。

图 4-8 确定策划书主题

策划书目标:

- **1. 增进同学间的友谊与合作**:通过团队项目,加强学生之间的沟通与协作能力。
- **2.** 提高**学生的身体**素质:通过各种运动项目,鼓励学生积极参与体育锻炼,提高健康水平。
- 3. **丰富校园文化生活**: 为学生提供一个展示自我、释放压力的平台,增加校园生活的多样性。
- **4. 培养团队精神和竞争意识**:通过比赛,培养学生的团队合作精神和健康的竞争意识。
- **5. 提高学生的组织和策划能力**: 让学生参与到运动会的策划和组织中,锻炼 他们的项目管理能力。
- 6. 增强学院的凝聚力:通过集体活动,增强学院内部的凝聚力和归属感。

人工智能基础与应用

"新小职",确定主题和目标后,你可以进一步细化策划书的内容,包括活动的具体项目、时间安排、场地布置、人员分工、预算计划等。如果你需要进一步的帮助,比如 具体的活动项目建议或者策划书的模板,随时告诉我。祝你的趣味运动会策划顺利!

根据目标和主题,撰写详细的活动方案,包括活动流程、时间安排、人员分工等。 利用 AI 工具进行文本润色和优化,使策划书更加专业和流畅。

步骤六:继续 Kimi 人机对话,输入:"请根据主题和目标,撰写详细的活动方案,包括活动内容、活动流程、时间安排、人员分工等",如图 4-10 所示。

基于你提供的主题"活力校园,运动无限"和目标,以下是一份详细的趣味运动 会活动方案:

活动名称:

活力校园,运动无限 —— 新乡职业技术学院趣味运动会

活动目的:

- 1. 增进同学间的友谊与合作。
- 2. 提高学生的身体素质。
- 3. 丰富校园文化生活。
- 4. 培养团队精神和竞争意识。
- 5. 提高学生的组织和策划能力。
- 6. 增强学院的凝聚力。

活动时间:

2024年12月20日 (星期六) 上午9:00 - 下午4:00

图 4-10

活动地点:

新乡职业技术学院操场及体育馆

活动对象:

新乡职业技术学院全体师生

活动内容:

- 1. 开幕式: 领导致辞、运动员代表宣誓。
- 2. 团队接力赛: 4x100米接力。
- 3. 趣味拔河:每队10人,三局两胜制。
- 4. 袋鼠跳: 穿着袋鼠装进行跳远比赛。
- 5. 两人三足: 两人一组, 绑腿赛跑。
- 6. 乒乓球接力: 用乒乓球拍托球跑, 球不掉地。
- 7. 跳绳比赛: 单人和团队跳绳。
- 8. 闭幕式及颁奖典礼: 颁发各类奖项。

图 4-11 趣味运动会活动方案

🤧 Kimi AI 生成的活动策划书

活动名称:

活力校园,运动无限 —— XX学校趣味运动会

活动目的:

- 1. 增进同学间的友谊与合作。
- 2. 提高学生的身体素质。
- 3. 丰富校园文化生活。
- 4. 培养团队精神和竞争意识。
- 5. 提高学生的组织和策划能力。
- 6. 增强学院的凝聚力。

活动时间:

20××年×月×日(星期六)上午9:00-下午16:00

活动地点:

XX学校操场及体育馆

活动对象:

××学校××学院(系)一年级学生

活动内容:

开幕式: 领导致辞、运动员代表宣誓。

团队接力赛: 4×100 米接力。

趣味拔河:每队10人,三局两胜制。

袋鼠跳:穿着袋鼠装进行跳远比赛。

两人三足:两人一组,绑腿赛跑。

乒乓球接力: 用乒乓球拍托球跑, 球不掉地。

跳绳比赛:单人和团队跳绳。

闭幕式及颁奖典礼:颁发各类奖项。

活动流程:

- 8: 30-9: 00: 参与者签到, 场地准备。
- 9:00-9:30:开幕式,领导致辞。
- 9:30-10:30:团队接力赛。
- 10:30-11:30:趣味拔河。
- 11: 30-13: 00: 午餐及休息。
- 13:00-14:00:袋鼠跳。
- 14:00-15:00:两人三足。
- 15:00-16:00:乒乓球接力。
- 16:00-17:00:跳绳比赛。



17:00-17:30: 闭幕式及颁奖典礼。

人员分工:

策划组:负责整体策划和协调。

宣传组:负责制作海报、横幅,通过社交媒体宣传。

场地组:负责场地布置和维护。

裁判组:负责比赛规则的解释和执行。

后勤组:负责餐饮、医疗等后勤保障。

记录组:负责记录比赛成绩和撰写新闻稿。

安全组:负责现场安全和紧急情况处理。

预算计划:

场地布置: 横幅、标志牌、彩带等。

器材租赁:接力棒、袋鼠装、跳绳等。

奖品: 奖杯、奖牌、证书。

其他: 医疗用品、打印费用等。

注意事项:

所有参与者必须遵守比赛规则。

确保所有活动安全第一,避免受伤。

保持场地清洁,活动结束后进行垃圾清理。

(三) AI 技术在策划书撰写中的应用

AI 技术在策划书撰写中的应用主要体现在以下 3 个方面:

文本生成:利用 AI 工具生成初稿,节省时间和精力。

文本润色: AI工具可以帮助优化文本,使其更加简洁、明了。

数据分析: AI 工具可以进行数据分析和预测, 为策划书提供科学依据。

三、跨学科交流中的AI应用

(一) 跨学科交流的重要性

跨学科交流是现代教育和科研的重要趋势之一。通过跨学科交流,可以促进不同领域的知识和技术的融合,推动创新和进步(见图 4-12)。

1. 推动科学发现和技术创新

AI 的应用正在改变科学研究的范式,通过跨学科合作,AI 技术被用于计算和预测分子和材料的性质,如蛋白质分子结构和高效太阳能电池材料。2024年诺贝尔物理学奖

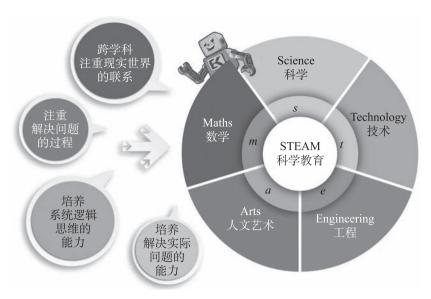


图 4-12 跨学科交流

颁给 AI 科学家,显示了 AI 在物理学等传统自然学科中的巨大潜力和跨学科研究的重要性。

2. 促进学科深度融合

AI 的发展不仅是技术问题,还涉及人类社会与文明的健康发展。AI 在具体场景中的成功应用建立在人机合作的基础上,依赖于人的隐性劳动,未来可能出现的人机融合将带来更多值得思考的问题。

3. 提升教育质量和研究水平

高等教育机构通过跨学科合作,更有效地培养适应未来社会需求的人才。AI 技术的引入不仅改变了教育模式,还带来了伦理和社会影响的深刻反思。

4. 解决复杂的社会挑战

跨学科合作在 AI 研究中的重要性日益凸显,尤其是在应对复杂的社会挑战时。不同学科的研究者在 AI 相关项目上可能会有相似的研究目标,但他们的视角和方法各不相同,通过汇聚这些不同的视角,能够激发出更具创新性和影响力的研究成果。

5. 培养高层次复合型人才

通过跨学科平台,可以建设新工科,服务新文科,实现人工智能对相关学科的赋能,探索"人工智能+X"高层次复合型人才培养新模式。

6. 促进社会责任和伦理考量

AI 的设计者、政策制定者及社会公众之间的对话是必要的,特别是在如何利用算法实现社会目标上。跨学科合作不仅能推动 AI 算法的进步,也为各行各业提供了更为平等和有效的解决方案。

7. 加速科学发现和智能化实验

新的技术和方法为基础科研带来了丰富的契机,加速了科学发现、促进了跨学科合作与创新、推动了智能化的实验与模拟。

(二) AI 在跨学科交流中的应用

AI 技术在跨学科交流中发挥着重要作用,主要体现在以下几个方面:

- (1) 自然语言处理: AI 可以帮助理解和翻译不同学科的专业术语, 促进跨语言交流。
- (2) 知识图谱: AI 可以构建知识图谱,帮助不同学科的研究人员快速找到相关的研究成果和资源。
- (3)智能推荐: AI 可以根据用户的兴趣和需求,推荐相关的学术论文、会议和专家,促进学术交流。

(三)案例项目: AI辅助"机器人"编程

在技能文化节上,机器人表演环节无疑是一大亮点。其中,AI 辅助机器人编程技术的应用尤为引人注目。计算机学院展示了一款集成了 AI 辅助编程功能的机器人。这款机器人不仅能够执行简单的任务,还能通过自然语言与人类进行交互。

AI 辅助机器人编程是指利用人工智能技术来辅助开发者进行机器人程序的设计、编写和调试。这种技术可以大大提高开发效率,降低编程难度,使得非专业人士也能参与到机器人编程中来。

"新小职"带你一起 AI 辅助 Python 语言进行机器人程序设计。例如: 当观众对机器人说"你好"时,机器人能够理解这一指令,并通过语音合成技术回应"你好"。

操作步骤以"讯飞星火"AI为例。

步骤一: 打开"浏览器", 输入 https://xinghuo.xfyun.cn。

步骤二:进入"讯飞星火"AI界面,开启对话。

步骤三:输入"你好,我是新乡职业技术学院的"新小职"。开启人机对话(见图 4-13)。

你好,有什么可以帮你吗?

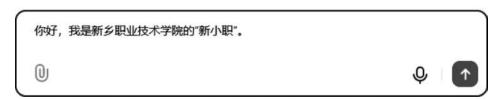


图 4-13 "讯飞星火"对话框

"讯飞星火"进行人机互动(见图 4-14)。

步骤四:输入"请帮我推荐 python 常用的机器人编程库。"如图 4-15 所示。

の訊だ星火

你好,"新小职"!很高兴认识你。

作为新乡职业技术学院的一员,你可能对学院的各个方面都有一定的了解和关注。如果你有任何关于学院的问题,比如课程设置、师资力量、校园活动等,都可以问我哦。我会尽我所能为你提供帮助和解答。同时,也欢迎你分享你在学院的学习生活经历和感受,让我们一起为学院的发展和进步贡献自己的力量。

图 4-14 "讯飞星火"人机互动

心讯达星火

使用工具: 联网搜索

Python在机器人编程领域拥有广泛的应用,其简洁的语法和强大的库支持使得它成为开发者的首选语言之一。以下是一些常用的Python机器人编程库:

- RoboMaster SDK: 这是一个基于Python开发的开源机器人编程框架,用于构建机器人应用程序。它集成了基于视觉、雷达、IMU等传感器的底层通信机制,能够让用户快速进行机器人项目的开发₽。
- 2. ROS (Robot Operating System): ROS是一个开源的机器人操作系统,提供了丰富的接口和工具,帮助开发人员开发功能更加强大的机器人应用程序。它的功能包括机器人状态建模、运动规划、通信、传感器融合、导航等户。
- 3. **Webots**: Webots是一款免费的机器人仿真软件,允许用户设计和模拟各种类型的机器人。它支持多种编程语言,包括Python,并且可以与ROS无缝集成P。
- 4. Movelt!: Movelt!是─个用于机器人运动规划的开源软件包,提供了 从简单到复杂的运动规划算法。它可以与其他ROS软件包一起使用,为 机器人提供高效的运动规划能力₽。
- 5. **PythonRobotics**: 这是一个专门用于机器人算法研究和模拟的Python 库,包含了多种经典的机器人算法实现,如路径规划、定位、导航、SLAM(同步定位与地图构建)等PPP。
- 6. **python-urx**:这是一个基于Python的开源库,专门用于与Universal Robots (优傲机器人)的协作机器人进行交互。它允许程序员通过 Python语言控制UR机器人的运动、读取传感器数据以及执行复杂的任务 P。
- 7. **PyRobotics**: PyRobotics是一个用于学习和研究机器人技术的Python 库,提供了多种机器人模型和控制器的实现,适用于教育和研究领域 ≥。
- 8. **RoboPy**: RoboPy是另一个流行的Python机器人编程库,提供了丰富的API和工具,帮助开发者快速构建和测试机器人应用程序 ≥。
- 9. **RoboPy2**: 作为RoboPy的升级版, RoboPy2提供了更多的功能和改进, 支持更多的机器人平台和传感器类型 ₽。

"新小职":以上库各有特点,适用于不同的机器人编程需求。在选择时,可以根据具体的应用场景和需求来决定使用哪个库。同时,由于机器人技术的快速发展,新的库和工具不断涌现,建议持续关注相关领域的动态,以便及时了解和应用最新的技术和工具。"新小职": 我选择使用 "RoboPy2 库"进行机器人编程。你可以试一试其他机器人库呦。

步骤五:继续 AI 对话,输入: "RoboPy2 能实现人机交互吗?"如图 4-16 所示。

心讯区星火

使用工具: 联网搜索

关于RoboPy2是否能实现人机交互,这主要取决于其设计、功能以及所支持的 库和接口。RoboPy2是一个基于Python的机器人编程库,它提供了丰富的API 来控制和管理机器人的各种行为,包括运动、传感器读取、任务执行等。然 而,具体到人机交互方面,这通常涉及到更高级的智能和感知能力,如语音识 别、面部表情识别、情感计算等。

从现有的搜索结果来看,RoboPy2本身并没有直接提及是否支持人机交互的具体功能。但是,考虑到Python生态系统中存在众多用于人机交互的库和工具(如TensorFlow, Keras, PyTorch等用于机器学习和深度学习; OpenCV用于计算机视觉; SpeechRecognition用于语音识别等), RoboPy2有可能通过集成这些库来实现一定程度的人机交互功能。

此外,人机交互不仅仅是技术层面的实现,还需要考虑用户体验、界面设计、 交互逻辑等多个方面。因此,即使RoboPy2具备实现人机交互的技术基础,也 需要开发者根据具体的应用场景和需求进行定制化的开发和优化。

图 4-16 RoboPy2 是否能实现人机交互

步骤六:输入"RoboPy2如何接收人类语音命令?"如图 4-17 所示。

心讯还星火

使用工具: 联网搜索

RoboPy2可以通过集成语音识别技术来接收人类语音命令。具体来说,可以利用科大讯飞的中文语音库来实现这一功能》。以下是实现步骤:

- 1. **下载并配置科大讯飞语音库**:需要从科大讯飞开放平台下载适用于Linux系统的SDK,并在本地环境中进行配置。这包括解压SDK文件、编译示例程序等步骤PP。
- 2. 编写语音识别代码:在RoboPy2的工作空间中,可以编写一个Python 脚本(如voice.py)来调用科大讯飞的语音识别接口。这个脚本会启动语音识别进程,捕获用户的语音输入,并将识别结果输出到一个文本文件中P。
- 3. 读取识别结果并处理:在另一个Python脚本(如 teleop_twist_keyboard.py)中,可以添加读取上述文本文件数据的代码。这样,当用户通过语音发出指令时,这些指令会被写入文本文件,然后由teleop_twist_keyboard.py脚本读取并根据指令内容控制机器人 移动 P。

"新小职":还可以考虑使用其他语音识别工具或库,如 PocketSphinx 等,这些工具也提供了 Python 接口,可以方便地集成到 RoboPy2 项目中。需要注意的是,由于语音识别技术涉及复杂的算法和模型训练,因此在实际应用中可能需要根据具体需求进行调整和优化。同时,为了确保机器人能够准确理解用户的语音指令,还需要考虑噪声干扰、口音差异等因素,并采取相应的措施进行优化。

步骤七:输入"编写语音识别'你好'的代码"如图 4-18 所示。

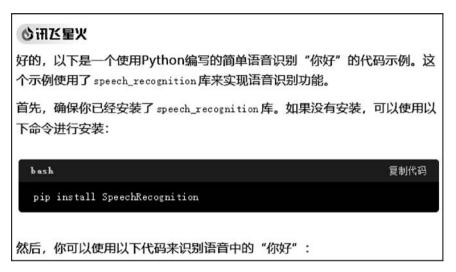


图 4-18 输入"编写语音识别'你好'的代码"

🥦 "讯飞星火" AI 给出的程序代码:

audio = recognizer.listen(source)

```
import speech _ recognition as sr

def recognize _ speech _ from _ mic(recognizer, microphone):
    """Transcribe speech from recorded from microphone."""
    if not isinstance(recognizer, sr. Recognizer):
        raise TypeError("recognizer must be speech _ recognition. Re cognizer instance")
    if not isinstance (microphone, sr. Microphone):
        raise TypeError("microphone must be speech _ recognition. Microphone instance")

with microphone as source:
    print("Adjusting for anbient noise...")
    recognizer. adjust _ for _ ambient _ noise(source)
    print("Listening...")
```

```
response = {
    "success": True,
    "error": None,
    "transcription": None
    try:
    response["transcription"] = recognizer.recognize_google(audi o, lan-
quage='zh-CN')
    except sr. RequestError:
      response["success"] = False
      response["error"] = "API unavailable"
    except sr. UnknownValueError :
      response["error"] = "Unable to recognize speech
    return response
    if _ _name _ - " _ _ main _ _ ":
      recognizer = sr. Recognizer()
      microphone = sr. Microphone()
      print("Say '你好'into the microphone...")
      response = recognize _ speech _ from _ mic(recognizer, microphone)
      if response["success"]:
        print("You said:{}".format(response["transcription"]))
        if response["transcription"]="你好":
          print ("Hello detected!")
        else:
          print("Didn't detect'你好'.")
          print("I didn't catch that. What did you say? Error: {}". for
mat (response["error"]))
```

步骤八:输入"请继续编写识别用户说你好后,回复'很高兴认识您,欢迎来到机器人的世界'"。

心讯还星火

好的,我们可以在之前的代码基础上进行扩展,当识别到用户说"你好"时,让机器人回复"很高兴认识您,欢迎来到机器人的世界"。以下是完整的代码示例:

图 4-19 编写机器人回复代码

```
bash 安装 SpeechRecognition、pyaudio 和 pyttsx3 库.
pip install SpeechRecognition pyaudio pyttsx3
python 代码
import speech recognition as sr
import pyttsx3
def recognize and respond():
   # 初始化识别器和语音合成引擎
   recognizer = sr. Recognizer()
   engine = pyttsx3.init()
    # 使用默认的麦克风作为音频源
   with sr. Microphone() as source:
       print("请说 '你好'...")
       # 调整麦克风的噪音水平
       recognizer. adjust for ambient noise(source)
       # 监听一段时间的音频数据
       audio = recognizer.listen(source)
   try:
       # 使用 Google Web Speech API 进行识别
       text = recognizer.recognize_google(audio, language='zh-CN')
       print(f"你说的是: {text}")
       # 检查是否识别到"你好"
       if "你好" in text:
           print("识别成功!")
          respond("你好")
       else:
           print("没有识别到'你好'.")
   except sr. Unknown Value Error:
       print("无法理解音频")
   except sr. Request Error as e:
       print(f"请求错误; {e}")
```

```
def respond(text):
    # 使用语音合成引擎回复文本
    engine = pyttsx3. init()
    engine. say(text)
    engine. runAndWait()

if _ _ name _ _ == "_ _ main _ _ ":
recognize _ and _ respond()
```

知识拓展

网络连接:由于 SpeechRecognition 库默认使用 Google Web Speech API 进行语音识别,因此需要有稳定的网络连接。

语言设置:在调用 recognize _ google 方法时, 我们设置了语言参数为 zh-CN, 表示中文。如果你需要识别其他语言,可以相应地修改这个参数。

环境噪声:为了提高识别的准确性,可以在安静的环境中运行此脚本,或者使用 recognizer.adjust_for_ambient_noise (source)来自动调整麦克风的噪音水平。

语音合成: pyttsx3 库支持多种 TTS 引擎,可以根据需要选择不同的引擎。

(四)人工智能编程语言在职业教育中的重要性

1. 贴合行业发展趋势,提升学生就业竞争力

随着人工智能技术在各个领域的广泛应用,市场对具备人工智能编程技能的人才需求日益增长。在职业教育中教授人工智能编程语言,能够让学生紧跟行业发展潮流。例如:在智能制造领域,企业需要技术人员能够运用相关编程语言对智能设备进行编程与维护;在金融行业,利用人工智能编程语言构建风险预测模型、开发智能投顾系统等也成为趋势。学生掌握这些编程语言后,在就业市场上会更具优势,能够轻松适应不同行业的技术岗位需求,获得更多的就业机会。

(1) 智能制造领域:工业机器人编程。

在智能制造行业,工业机器人的应用越来越广泛(见图 4 - 20)。很多知名机器人厂商的设备在工厂中承担着诸如焊接、装配、搬运等重要任务。这些工业机器人的编程控制,很多都依赖于人工智能编程语言。



图 4-20 工业机器人

知识拓展

"新小职"是一名机电一体化专业学生,他在校期间学习了 Python 语言以及相关的机器人编程库。毕业后,"新小职"进入一家汽车制造企业。在企业的生产线上,有大量的工业机器人负责汽车零部件的焊接工作。"新小职"利用所学的 Python 编程知识,对机器人的运动轨迹进行优化。他通过编写程序,让机器人能够更精准、高效地完成焊接任务,不仅提高了焊接质量,还将生产效率提升了 20%。凭借这一技能,"新小职"在企业中迅速得到重用,薪资待遇也十分可观。同时,随着智能制造行业的持续发展,像他这样掌握人工智能编程技能的人才,在行业内的就业机会不断增加,无论是在传统制造业的智能化升级过程中,还是新兴的智能制造企业中,都备受青睐。

(2) 金融科技领域:量化交易策略开发。

在金融行业,量化交易已经成为重要的发展方向。量化交易通过数学模型和计算机 算法来实现交易决策,这其中离不开人工智能编程语言。

知识拓展

"新小职"是金融专业的学生,他在学习过程中接触到了Python语言,并深入 学习了量化交易相关的知识。毕业后,"新小职"进入一家金融科技公司。公司希 望开发一套新的量化交易策略,以适应复杂多变的金融市场。"新小职"运用 Python 语言,结合金融市场的历史数据和各种技术指标,构建了量化交易模型。通过 不断地优化和回测, 他开发的交易策略在实盘交易中取得了良好的效果, 为公司带 来了显著的收益。在金融科技行业,具备人工智能编程能力的金融专业人才,能够 更好地将金融知识与技术相结合,开发出创新的金融产品和服务。"新小职"也因 为自己的技能,在行业内拥有了更多的职业发展机会,不仅可以在金融科技公司从 事量化交易相关工作,还可以在银行、证券等传统金融机构的金融科技部门大展 身手。

表 6-2

分类	详情
行业背景	金融市场的快速发展和交易的日益复杂,使得量化交易成为金融科技领域的关键发展方向。传统的人工交易决策受限于人的精力和分析能力,难以在瞬息万变的市场中及时捕捉机会和控制风险。量化交易借助数学模型和计算机算法,能够对海量金融数据进行快速分析和处理,实现更精准的交易决策。
编程应用	在量化交易策略开发中,Python 语言凭借其丰富的数据分析库(如 Pandas、NumPy)和机器学习库(如 Scikit-learn),成为主流的编程语言。例如,利用 Pandas 可以高效地处理和分析金融市场的历史数据,包括股票价格、成交量等;通过 NumPy 进行数值计算,为模型构建提供基础支持;借助 Scikit-learn 中的机器学习算法,如线性回归、决策树等,构建量化交易模型,预测市场走势和资产价格波动。
学生学习实践	"新小职"在课程设置中融入了 Python 编程和量化交易相关课程。首先学习 Python 语言的基础语法和数据结构,掌握基本的编程能力。随后,深入学习量化交易的理论知识,包括交易策略的设计原理、风险评估方法等。在实践环节,学生们利用金融数据模拟交易平台,运用所学的 Python 知识编写量化交易策略代码。通过不断地调试和优化,让策略在模拟交易环境中进行测试和验证,积累实践经验。
就业优势	像"新小职"这样掌握 Python 编程用于量化交易策略开发的金融专业学生,在就业市场上极具竞争力。毕业后进入金融科技公司,"新小职"能够迅速投入到量化交易策略的开发工作中。他开发的新策略在实盘交易中为公司带来显著收益,得到公司的高度认可。在金融科技行业,这类人才既懂金融知识又具备编程能力,不仅可以在金融科技公司从事量化交易策略研发、交易系统维护等工作,还能在银行、证券等传统金融机构的金融科技部门,参与智能投顾系统开发、风险评估模型构建等重要项目,职业发展前景广阔。

2. 培养创新思维与实践能力

人工智能编程语言的学习过程充满挑战与创新。在职业教育中,学生通过实际项目

操作,如使用 Python 进行图像识别项目开发、用 JavaScript 搭建简单的人工智能交互界面等,不断尝试新的算法和模型,从而激发他们的创新思维。同时,在解决实际问题的过程中,学生需要将理论知识转化为实际操作,这大大提升了他们的实践动手能力。比如在开发智能聊天机器人的实践中,学生不仅要掌握自然语言处理的理论知识,还要运用编程语言将其实现,从数据预处理到模型训练,每一个环节都锻炼了他们的实践能力。



智能舆情分析系统开发案例

1. 项目背景

在当今信息爆炸的时代,企业和政府部门对于舆情的监测和分析需求日益迫切。传统的人工舆情监测方式效率低下,且难以全面、准确地把握舆情动态。某职业院校计算机专业的学生,在学习了Python语言以及相关的人工智能技术后,决定开发一个智能舆情分析系统。

2. 实践过程:

- (1) 数据收集阶段:运用 Python 的网络爬虫技术,从各大新闻网站、社交媒体平台等收集相关的文本数据。同学们需要不断调试爬虫程序,以应对不同网站的反爬虫机制,锻炼同学们解决实际问题的实践能力。
- (2) 数据预处理:利用 Python 的自然语言处理库(如 NLTK、jieba) 对收集到的文本数据进行清洗、分词、去停用词等处理,将原始文本转化为适合分析的格式。
- (3) 模型构建:采用机器学习算法(如朴素贝叶斯、支持向量机)和深度学习模型(如 LSTM),通过 Python 的 Scikit-learn 和 TensorFlow 库进行模型的搭建和训练。同学们需要尝试不同的模型参数和组合,以找到最适合舆情分析的模型。
- (4) 系统开发:将训练好的模型集成到一个 Web 应用程序中,使用 Python 的 Flask 框架搭建后端,实现数据的存储、处理和分析结果的展示。前端则使用 HT-ML、CSS 和 JavaScript 进行页面设计,提供友好的用户交互界面。
- (5) 创新体现: 在项目开发过程中, 创新性想法来提高舆情分析的准确性和时效性。例如, 引入了情感分析和主题分类相结合的方法, 不仅能够判断舆情的正负倾向, 还能对舆情的主题进行分类, 让用户更清晰地了解舆情的焦点。尝试将实时数据处理技术应用到系统中, 实现对舆情的实时监测和预警。
- (6) 收获成果:通过项目锻炼,将所学的人工智能编程语言知识应用到实际项目中,提升实践能力,培养创新思维。

3. 促进跨学科融合,拓宽学生知识视野

人工智能编程语言涉及到多个学科领域的知识, 如数学、统计学、计算机科学等。

在职业教育中学习这些编程语言,有助于促进跨学科融合。以预测学生成绩的案例来 说,学生需要运用数学知识理解机器学习算法背后的原理,同时结合统计学知识对数据 进行分析和评估,再利用 Python 等编程语言实现整个模型的构建。这种跨学科的学习 方式,能够拓宽学生的知识视野,让他们从多个角度去思考和解决问题,培养综合性的 职业素养。

4. 助力职业教育课程改革与创新

人工智能编程语言的引入为职业教育课程改革提供了新的方向和动力。传统的职业 教育课程可能侧重于单一技能的培养,而人工智能编程语言的融入,使得课程更加多元 化和现代化。学校可以开设相关的人工智能编程课程,如"Python 人工智能基础""人 工智能与数据分析"等,丰富课程体系。同时,在其他专业课程中也可以融入人工智能 编程的元素,如在机械制造专业课程中,加入利用人工智能编程语言控制智能机床的内 容,让学生在学习专业知识的同时,掌握前沿的技术,提升职业教育的质量和吸引力。



1. 项目背景

在体育领域, 为了提高运动员的竞技水平, 需要对运动员的动作进行精细分 析,找出潜在的问题并提供优化建议。例如:在田径短跑项目或体操项目中,精准 分析运动员的动作细节可以帮助教练和运动员改进技术(见图4-21)。



图 4-21 对运动员的动作进行精细分析



1. 运动人体科学专业知识

运动人体科学专业人员对人体解剖学、运动生理学和生物力学等知识有深入了解。他们通过高速摄像机等设备采集运动员的动作数据,然后运用专业知识分析运动员的关节角度、肌肉发力情况、身体重心转移等关键要素。这些分析结果可以帮助确定运动员动作的优点和不足,为后续的优化提供依据。例如:在短跑运动员的起跑动作分析中,他们可以确定运动员腿部肌肉的发力顺序是否合理,起跑姿势是否有利于快速加速。

2. 计算机视觉专业知识

计算机视觉专业人员利用 Python 语言结合 OpenCV 等计算机视觉库,对采集 到的运动员动作视频进行处理。他们可以实现目标检测(识别运动员的身体部位)、 动作跟踪(跟踪运动员身体部位在视频中的运动轨迹)等功能。通过计算机视觉算 法,将视频中的动作信息转化为可以量化的数据,为后续的分析和优化提供数据 支持。

3. 数据科学与人工智能专业知识

数据科学和人工智能专业人员使用 Python 语言和相关的机器学习库(如 Sci-kit-learn)或深度学习框架(如 TensorFlow)。他们将计算机视觉获取的量化动作数据与运动人体科学的分析结果相结合,构建运动员动作分析模型。例如,通过机器学习算法,根据大量优秀运动员的动作数据和成绩数据,建立动作特征与运动成绩之间的关系模型。利用这个模型可以对运动员的现有动作进行评估,并通过深度学习中的强化学习算法,为运动员提供动作优化建议,帮助运动员找到最佳的动作策略。

4. 知识融合与成果

运动人体科学提供了动作分析的专业理论基础, 计算机视觉将动作视频转化为可用的数据,数据科学与人工智能则基于这些数据构建分析和优化模型。通过这种跨学科融合, 开发出的运动员动作分析与优化系统能够帮助教练和运动员更科学地评估动作技术, 提供个性化的训练建议, 从而有效提高运动员的竞技水平。

(五) 人工智能辅助编程

1. 人工智能代码生产的核心优势

(1) 提高开发速度。

人工智能代码生成工具可以快速生成代码片段,减少开发者手动编码的工作量,从 而加速开发流程。例如: GitHub Copilot 可以根据开发者的注释和上下文自动生成整行 或整个函数的代码,使开发者能够更快地完成编码任务。

(2) 减少错误。

AI 生成的代码通常经过优化,能够减少人为错误。例如: DeepCode 可以通过机器 学习算法自动识别代码中的错误、漏洞和潜在风险,并提供优化建议,帮助开发者提高 代码质量。

(3) 促进学习。

对于新手开发者,人工智能代码生成工具可以作为学习和探索新代码模式的工具。 通过观察 AI 生成的代码,新手开发者可以快速了解不同编程语言的语法和编程技巧, 加速学习过程。例如: GitHub Copilot 可以帮助开发者快速学习和探索不熟悉的编程语 言或库。

(4) 代码一致性。

AI 可以生成风格一致的代码,提高代码库的整体质量。例如: Cursor 可以根据当 前项目代码的背景生成更合适的建议,并自动生成文档注释,使代码更加规范和易于阅 读,便于团队合作。

2. 人工智能代码生成的实施步骤

步骤——选择 AI 代码生成工具:选择适合项目需求的 AI 代码生成工具,如 GitHub Copilot、DeepCode、Replit 等。

步骤二——集成开发环境:将 AI 代码生成工具集成到常用的 IDE (集成开发环境) 中。例如: GitHub Copilot 集成在 Visual Studio Code 编辑器中。

步骤三——提供上下文信息:为 AI 模型提供足够的上下文信息,以生成准确的代 码片段。例如: 开发者可以输入注释或部分代码来描述他们希望代码执行的操作。

步骤四——评估和调整:评估生成的代码的正确性和适用性,并根据需要进行调 整。对于一些高度自定义的业务逻辑,生成的代码可能需要进行人工调整。

3. 人工智能代码生成的应用场景

自动补全和代码建议: AI 代码生成可以作为 IDE 中的自动补全和代码建议工具, 帮助开发者快速编写代码。例如: Replit 的 AI 编程助手 Ghostwriter 使用 GPT 模型提 供代码建议、生成代码、自动修复等功能。

重构和代码优化: AI 可以帮助识别代码中的潜在问题,并提出优化建议或自动重构 代码。例如: DeepCode 提供高质量的代码质量提升建议。

快速原型开发:在原型开发阶段,AI可以快速生成基础代码框架,加速开发过程。 例如:腾讯的"AI 代码"支持根据需求生成完整的代码框架,帮助开发者节省时间。

4. 人工智能代码生成的工具推荐

GitHub Copilot:集成在 Visual Studio Code 编辑器中,支持多种编程语言,能够根 据注释和上下文自动生成代码。

Cursor: 特别关注在代码的上下文中进行优化和提升,支持多种编程语言,提供代 码补全、自动修复、生成文档等功能。

Replit: 在线集成开发环境,支持多种编程语言,结合内建的 AI 编程助手 Ghostwriter,提供代码建议、生成代码、自动修复等功能。

DeepCode:由 Snyk 提供,使用 AI 模型分析代码中的潜在问题、漏洞和性能瓶颈,并提供优化建议。

CodeT5:由 Salesforce 开发,支持代码生成、代码补全、错误修复等功能,特别擅长处理自然语言与代码之间的转换。

JupyterLab AI: 专为数据科学家、机器学习工程师设计,提供代码补全、自动化分析和优化功能。

腾讯 AI 代码:结合深度学习和大数据分析,提供全面的智能编程支持,包括高效代码补全、智能错误检测、代码自动化生成和集成调试功能。

科大讯飞 iflycode 开发助手:基于语音识别与自然语言处理技术,能够通过语音输入快速生成代码,适合语音交互频繁的开发环境(见图 4-22)。



图 4-22 iflyCode 智能编程助手



AI 技能提升

编写代码

Cursor 是一款基于外部大语言模型的 AI 代码编辑器,类似于支持生成式 AI 聊天机器人的模型,它提供了代码补全、解释、重构和基于自然语言描述的完整函数生成等功能。

假设你需要编写一个 JavaScript 函数,用于检查一个数组是否包含某个元素。你可以编写如下代码:

```
function contains(array, element) {
    //...}

Cursor 会自动提供补全建议,生成如下代码:
function contains(array, element) {
    return array. includes(element);}

同时, Cursor 会生成文档注释:
/**

* 检查数组是否包含某个元素

* @param {Array} array-要检查的数组

* @param {* } element-要检查的元素

* @returns {boolean}-是否包含该元素

* /function contains(array, element) {
    return array. includes(element);}
```



数据处理

假设你需要从一个嵌套的数组中提取名字,并将其存储在一个新的数组中。你可以编写如下注释:

```
// 遍历对象数组的数组以转换数据// 示例:从数据数组中提取名字// 期望结果:
['John', 'Jane', 'Bob']const data = [
        [ { name: 'John', age: 25 }, { name: 'Jane', age: 30 }],
        [ { name: 'Bob', age: 40 }]];
        然后按 Tab 键, GitHub Copilot 会生成如下代码:
        const mappedData = data. flatMap(sublist => sublist. map(person => person. name)); console. log(mappedData); // 结果: ['John', 'Jane', 'Bob']
```



AI 技能提升

API 调用

假设你需要编写一个函数,用于调用一个 API 并处理响应。你可以编写如下注释:

```
# 调用 API 获取用户信息# API URL: https://api.example.com/users# 返回
值:用户信息列表
   然后按 Tab 键, GitHub Copilot 会生成如下代码:
   Python 复制
   import requests
   def get users():
       url = "https://api.example.com/users"
       response = requests.get(url)
       if response. status \_ code ==200:
          return response. json()
       else:
          return []
```



AI 技能提升

文件操作

假设你需要编写一个函数,用于读取一个文件并返回其内容。你可以编写如下 注释:

读取文件内容# 文件路径:/path/to/file.txt# 返回值:文件内容 然后按 Tab 键, GitHub Copilot 会生成如下代码: def read _ file(file _ path): with open(file_path, 'r') as file: return file. read()



一、单选题

- 1. 以下哪种编程语言在人工智能领域应用广泛, 且以简洁的语法和丰富的库支持著 称? ()

 - A. Java B. Python C. C++ D. Go
- 2. 在使用人工智能辅助编程工具时,以下哪个步骤是生成高质量代码的关键?()
 - A. 选择合适的编程语言
- B. 提供详细的上下文信息