



AI 赋能通识教育
精品系列

全国高等学校计算机教育研究会
优秀教材一等奖

计算 与人工智能 概论

国家级一流本科课程配套教材
紧跟最新技术发展，助力应用人才培养
适合零基础读者学习使用

第2版
微课版

罗娟◎主编
刘璇 蔡宇辉 贺再红 李小英 陈娟◎编著

- 微课视频
- PPT 课件
- 教学大纲
- 习题答案
- 在线实训
- 模拟试卷



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS

AI 赋能通识教育
精品系列

计算 与人工智能 概论

第2版
·
微课版

罗娟◎主编

刘璇 蔡宇辉 贺再红 李小英 陈娟◎编著



人民邮电出版社
北京

图书在版编目(CIP)数据

计算与人工智能概论：微课版 / 罗娟主编；刘璇
等编著. -- 2 版. -- 北京：人民邮电出版社，2025.
(AI 赋能通识教育精品系列). -- ISBN 978-7-115-67565-

1

I. TP18

中国国家版本馆 CIP 数据核字第 2025DV7463 号

内 容 提 要

本书从计算思维的角度出发，以人工智能相关问题为引导，在解决实际案例问题的过程中植入知识点，为各专业的学生在今后设计、构造和应用各种计算系统，求解本学科的问题奠定基础。全书内容分为计算与计算思维、程序设计与算法、人工智能与智能计算、网络与大数据 4 个部分，共 12 章，包括计算与人工智能概述、计算系统、Python 编程基础、Python 编程进阶、算法设计、智能感知、机器学习、智能决策、智能机器人、互联网信息处理、数据管理与大数据、数据分析。

本书适用于高等学校一年级新生的计算机导论等信息技术类基础课程，可作为高等学校计算机基础课程的教材，也可作为计算与人工智能概论相关课程的教材，还可作为计算机基础培训的教材和自学参考书。

◆ 主 编 罗 娟	
编 著 刘 璇 蔡宇辉 贺再红 李小英 陈 娟	
责任编辑 许金霞	
责任印制 沈 蓉 胡 南	
◆ 人民邮电出版社出版发行	北京市丰台区成寿寺路 11 号
邮编 100164	电子邮件 315@ptpress.com.cn
网址 https://www.ptpress.com.cn	
北京市艺辉印刷有限公司印刷	
◆ 开本：787×1092 1/16	
印张：17.75	2025 年 7 月第 2 版
字数：525 千字	2025 年 7 月北京第 1 次印刷

定价：69.80 元

读者服务热线：(010)81055256 印装质量热线：(010)81055316

反盗版热线：(010)81055315

广告经营许可证：京东市监广登字 20170147 号

前言

❖ 写作背景

当前信息技术日新月异，人工智能、物联网、云计算、大数据等技术，在社会经济、人文科学、自然科学等众多领域引发了一系列革命性的突破。数字时代对人才的需求不断变化，计算机和人工智能通识课程在内容建设与教学方法上面临着新的挑战。

本书作为大学生进入高校的第一门信息技术类基础课程的教材，从计算思维的角度出发，以人工智能相关问题为引导，通过线上线下混合教学的方式，以及任务驱动的案例化教学手段，帮助学生在解决实际任务的过程中学习知识，培养计算思维和人工智能实践能力，并为各专业的学生在今后设计、构造和应用各种计算系统，求解本学科的问题奠定理论基础。

本书满足数字时代对大学计算机基础课程改革的需要，符合未来智能社会对高质量人才的需求，参照教育部《高等学校人工智能创新行动计划》和《中国教育数字化战略行动》，按照新课程内容体系设计整体框架，从计算思维的角度出发，融入大量人工智能案例，同时深入浅出地讲解知识点，适合各专业的学生使用。

❖ 本书特点

按照初学者学习计算思维与人工智能的特点和规律，编者对本书各章的内容进行了精心安排。概括起来，本书具有如下特点。

（1）以“案例驱动，知识植入，计算思维，专业融合”为导向，优化内容，将计算思维与人工智能相结合。每章均围绕人工智能相关的案例，将知识点融入求解问题的过程中，以提高学生的学习兴趣，培养学生分析问题、解决问题的能力，强化学生的计算思维与人工智能技术应用能力，真正达成人工智能赋能教育的目的，帮助学生具备人工智能时代各类专业科学研究和实际工作所需要的能力。

(2) 深入挖掘课程教学价值,助力高校培养杰出人才,将不同专业人工智能的应用特点和价值理念有机地融入相关案例。本书通过宣传社会主义核心价值观,强化学生工程伦理教育,培养学生精益求精的工匠精神、科技兴国的民族使命感和理论联系实际的能力。本书利用通识课受众多、影响广的优势,发挥育人作用,将价值塑造、知识传授和能力培养融为一体。

(3) 因材施教,服务于专业教学。本书的案例和实验设计跨越多个专业领域,体现了多学科思维的融合,例如,将计算思维、程序设计与人工智能三者深度融合,强调在专业问题的求解中提炼核心知识,达到以练促教的目的;同时,体现了人工智能赋能专业在产业技术与学科理论融合、跨专业能力融合、多学科项目实践融合等方面的特点。

(4) 配套丰富的课程教学案例库。本书结合各专业相关课程的教学需求,适应新工科、新文科等发展趋势,可以促进学生提升交叉融合计算思维与各专业相关问题的能力。本书以智能驱动、思维跃迁为主线,引导学生重视问题的解决方法与过程,进而领悟人工智能的作用与意义,帮助教师避免在低年级人工智能教学实施过程中出现概念化、理论化的现象。

❖ 本书内容

本书共 12 章,分为计算与计算思维、程序设计与算法、人工智能与智能计算、网络与大数据四大部分。

第一部分为计算与计算思维,包括第 1~2 章的内容。

第 1 章为计算与人工智能概述。通过智能移动机器人路径规划案例,引出计算的概念、计算的历史、计算系统的组成及人工智能的发展等内容。

第 2 章为计算系统。以宇宙探测、智慧城市和智慧农业三个场景为例,描述计算环境和计算执行,包括并行计算、嵌入式计算、文件系统、未来的计算机与人工智能等内容。

第二部分为程序设计与算法,包括第 3~5 章的内容。

第 3 章为 Python 编程基础。通过机器人投篮的案例,讲述问题求解的计算思维方法、编程的基本概念、Python 语言的模块化程序构造、基本的程序设计方法等内容。

第 4 章为 Python 编程进阶。从机器人投篮案例任务分析出发,讲述 Python 语言的序列数据处理、映射数据处理和文件操作等内容。

第 5 章为算法设计。以火星探测为例,描述算法概念、算法时间复杂度分析、问题求解策略等内容。

第三部分为人工智能与智能计算,包括第 6~9 章的内容。

第 6 章为智能感知。通过自然语言生成的案例,描述自然语言处理、机器视觉、模式识别、大模型等人工智能在机器感知层面的重要研究内容。

第 7 章为机器学习。介绍监督学习、无监督学习和半监督学习这三种机器学习的方法。

第 8 章为智能决策。介绍基于搜索的最优路径决策、强化学习、群体智能这三种智能决策的实现方式。

第 9 章为智能机器人。介绍机器人技术、人机交互发展历程、人机交互的方式,以及人机交互的未来发展趋势等内容。

第四部分为网络与大数据，包括第 10~12 章的内容。

第 10 章为互联网信息处理。以即时通信和网络爬虫应用为例，介绍网络信息获取基础、网络爬虫与信息提取、搜索引擎工作原理和网络安全等内容。

第 11 章为数据管理与大数据。介绍常用的数据文件格式、计算机数据管理、数据库系统和大数据技术等内容。

第 12 章为数据分析。通过气候数据分析案例，介绍常用的数据分析方法和软件，数据采集、分析处理及其可视化等内容。

❖ 第二版修订内容

本书第一版自出版以来，被本科、高职高专等高校选作教材，用书单位反馈内容新颖、实用，与行业需求紧密相关。模块化的教学设计使得教师可以根据课程需要和学生特点，灵活选择教学内容，提高了教学的针对性和有效性。同时，书中丰富的实践操作和技能训练环节，极大地激发了学生的学习兴趣 and 动手能力。

第二版修订主要针对近三年国内外人工智能技术的发展，以及教育部人工智能推动人才培养范式改革，和人工智能通识课程体系的需求，对人工智能相关内容进行了增补和改进。同时注重产教融合，引入了华为、百度等企业最新的人工智能技术产品应用。第二版的配套数字资源更为丰富，线上视频、图文多种形式资源融合，更新了微课和课程思政案例库等，提供了更多元化的教学内容。

具体修订内容如下。

（1）针对人工智能近几年的飞速发展，增加“1.3.1 人工智能概述”一节，详细介绍了弱人工智能与强人工智能各自的特点，并分析了人工智能算法、算力与数据之间的关系。在“数据管理与大数据”中介绍了基于人工智能技术的新一代数据库系统的核心架构及其应用场景。在“互联网信息处理”中介绍了基于人工智能模型改进的 PageRank 算法，以及基于“感知—分析—决策”架构实现的人工智能网络防护系统。

（2）面向产教融合，在多个案例中引入“人工智能解决方案”，主要介绍多个人工智能平台，例如：“华为昇腾平台”的全栈架构，及其四个核心层次（即硬件层、异构计算架构层、AI 框架层和应用使能层）的特点和相互关系。

（3）针对大模型的发展应用，在“自然语言处理与大模型”中增加了“大模型及其应用”的内容，主要阐述大模型的发展历史、基本原理，并以华为盘古大模型为例，介绍大模型的使用方法及其调用接口。

（4）更新和优化人工智能案例，例如：“智能决策”的“案例分析与实现”中将基于蚁群算法的旅行家问题求解案例修改为基于华为 MindSpore 强化学习框架的机器人寻宝问题案例。

（5）根据修订内容对各章习题进行了全面更新，重点增加了人工智能相关的习题。

❖ 学时建议

教师可根据本校课程的学时情况，选择性地讲解本书重点章节（若学时充足，也可讲

解全部章节);此外,可以采用线上线下混合教学的方式开展本课程的教学工作。

本书各章的建议学时参见学时建议表,具体实施方案由各校根据实际教学计划确定。

学时建议表

章序	章名	课堂学时	实验学时
1	计算与人工智能概述	2~4	0
2	计算系统	2~6	2~4
3	Python 编程基础	4~6	3~5
4	Python 编程进阶	4~6	3~5
5	算法设计	8~10	4~8
6	智能感知	4~6	2~4
7	机器学习	4~6	4~8
8	智能决策	4~6	4~8
9	智能机器人	2~4	2~4
10	互联网信息处理	2~6	2~6
11	数据管理与大数据	2~6	2~6
12	数据分析	4~8	4~8
学时总计		42~74	32~66

❖ 配套资源

本书配套丰富的数字化教学资源,读者可以通过人邮教育社区(www.ryjiaoyu.com)或头歌在线实践教学平台下载本书配套的电子教案、慕课视频、实践素材、例题源程序、课后习题答案等。

❖ 编者团队

本书由罗娟主编,第一部分由罗娟编写,第二部分由贺再红、蔡宇辉编写,第三部分由刘璇、蔡宇辉编写,第四部分由陈娟、李小英、罗娟编写。

感谢湖南大学信息科学与工程学院和湖南大学教务处对本书编写出版的大力支持,特别感谢许多同行和朋友在本书撰写之初所提出的宝贵建议。感谢头歌在线实践教学平台对“计算与人工智能概论”课程实践项目的支持。

❖ 联系我们

限于编者水平,书中难免存在疏漏之处,敬请同行和广大读者批评指正。编者的邮箱为 juanluo@hnu.edu.cn,欢迎读者来函交流教学问题或索取本书配套资源。

罗娟

2025年5月于湖南大学

推荐序

数字化因可编程通用电子计算机的发明而起并迅速展开。经过 70 多年的发展，人类社会的数字化程度日益深化，并实现网络化，且正在向智能化的高级阶段迈进。计算无处不在，软件定义一切，“人机物”三元融合、万物智能互联的泛在计算时代正在到来。每一个接受高等教育的大学生不仅仅是现代数字化社会的直接受益者，更应该是未来数字文明的直接创造者。大学计算机通识教育应时而生，顺势而变。

我国大学计算机通识教育发端于 20 世纪 80 年代，其时代背景是计算机由特殊领域的高端专用计算工具变成服务更多工程领域的通用计算工具。这个阶段计算机通识教育的目的是帮助更多的理工科大学生学会使用计算机，此类教学工作一般由大学计算中心的教辅人员承担，教学活动在计算中心进行，其教学形态更接近实验课程。这是我国大学计算机通识教育的“第一台阶”。到 20 世纪 90 年代中后期，随着个人计算机的普及，依托计算中心开展的以使用计算机为教学目的的计算机通识教育失去了必要性，越来越多的学科专业希望学生具备应用计算机解决本学科领域特定问题的能力，计算机通识教育需要升级，其与专业教育的关系问题更加直接地显现出来。首先是教学内容问题。计算机应用技术是计算机科学与技术一级学科下的二级学科，计算机应用技术专业的本科专业课教学内容通常需要数百学时，如何在通识教育的几十学时内完成学生计算机应用能力的培养是教学内容问题。其次是教学方法问题。如何浓缩计算机应用技术专业教学内容，采用什么样的教学方法才能比较好地达成教学目的是教学方法问题。最后是教学任务问题。计算机应用开发能力的通识教育可能比专业教育更具挑战性，需要更多高水平的计算机专业教师承担教学任务，因此师资成为现实问题。

21 世纪前 10 年，国际计算机教育界就计算思维达成共识，试图将沉淀在计算机科学技术中的计算思维提取出来，传授给普通大学生。但计算机科学与技术学科毕竟是构造性学科，建立计算思维离不开系统性的实践认知，通过几十学时的通识授课构建计算思维概念是极具挑战性的工作。过去 20

年中，我国计算机专业的教育工作者锐意进取，探索“案例+实践”的计算思维教学内容和
方法改革，帮助学生培养自主学习新方法、解决新问题的能力和信心，取得了积极成效，
推动我国计算机通识教育从工具性通识教育向思维性通识教育的转变，我国计算机通识教
育跃上“第二台阶”。当前，计算机通识教育面临新的更大挑战。在“人机物”三元融合、
万物智能互联的泛在计算时代背景下，以计算机为中心的计算思维不足以覆盖泛在计算中
新的思维理念，如连接思维、大数据思维、演化思维、复杂网络思维等，这些思维理念超
越了经典计算思维的问题边界。应对挑战，需要我国计算机教育工作者付出更大的努力。

撰写本书的湖南大学教学团队为此开展了卓有成效的工作。这是一个年轻的计算机专
业教学团队。她（他）们以人工智能、机器人、物联网等泛在计算新兴应用领域的案例为
基础，不仅将经典的计算思维概念融入案例教学，还传递了连接、大数据等新的计算思
维。特别值得强调的是，她（他）们把教学工作置于开放的教学生态中，在传统教程和课
堂授课的基础上同步建设了大规模开放在线课程（Massive Open Online Courses，
MOOC）和大规模开放在线实践（Massive Open Online Practice，MOOP）资源，让
这门通识课程在开放的网络生态环境中演化成长，这种教学过程本身也向学生传递了复杂
系统生态的思维理念。

希望读者不仅仅是将本书视为传统出版物，更是将其视为一个成长演化的学习生态的
入口，进而通过本书加入“计算与人工智能”的学习生态之中。未来的大学计算机通识教
育将不限于传统课堂和实验室，也不限于大学的课程学习周期，其将以学习生态的形式伴
随每一个学习者的终身学习。我以为，这一努力将推动我国计算机通识教育迈向“第三台
阶”——学习生态台阶。



王怀民

国防科技大学教授

中国科学院院士

目录

第一部分 计算与计算思维

第1章

计算与人工智能

概述

1.1 计算的概念	2
1.1.1 什么是计算	2
1.1.2 图灵机	3
1.1.3 什么是计算机	4
1.1.4 算法、计算机语言与程序	8
1.2 什么是计算思维	12
1.2.1 计算思维的概念	12
1.2.2 问题求解方法	12
1.2.3 算法——计算系统的灵魂	15
1.3 探索人工智能	16
1.3.1 人工智能概述	16
1.3.2 智能移动机器人路径规划	17
1.3.3 人工智能历史	21
1.3.4 人工智能相关研究	23
1.3.5 人工智能解决方案	24
1.4 人工智能发展	25
1.4.1 人工智能应用领域	25
1.4.2 人工智能发展	28

习题	30
----	----

第2章

计算系统

2.1 计算与计算机简述	31
2.1.1 图灵机模型	34
2.1.2 计算机的工作原理	36
2.1.3 并行计算	40
2.1.4 嵌入式计算	42
2.2 计算机技术	45
2.2.1 编译、链接和装载程序	48
2.2.2 程序、进程和线程	52
2.2.3 文件系统	55
2.3 未来的计算机与人工智能	57
2.3.1 未来计算机的发展	57
2.3.2 未来人工智能的发展	60
习题	61

第二部分 程序设计与算法

第3章

Python 编程基础

3.1 问题求解的计算思维方法	63
3.1.1 问题分解	63

3.1.2	模式识别	63
3.1.3	抽象	64
3.1.4	算法设计	65
3.2	编程的基本概念	65
3.2.1	程序的基本要素	66
3.2.2	Python 语言	66
3.2.3	第一个 Python 程序：计算 篮球某时刻的坐标值	69
3.2.4	Python 基本语法	70
3.2.5	计算并绘制坐标点	73
3.3	模块化编程——函数	74
3.3.1	内置函数	74
3.3.2	自定义计算坐标的函数	75
3.3.3	调用函数绘制坐标点	76
3.3.4	函数的参数	76
3.3.5	函数返回	78
3.4	基本的程序设计方法—— 分支	78
3.4.1	单分支——if 语句	78
3.4.2	二分支——if-else 语句	78
3.4.3	多分支——if-elif-else 语句	79
3.4.4	智能型计算坐标函数	80
3.4.5	在合理范围内绘制 坐标点	81
3.5	基本的程序设计方法—— 循环	82
3.5.1	while 循环	82
3.5.2	在合理范围内绘制多个 坐标点	83
3.5.3	循环嵌套	84
3.5.4	在合理范围内绘制篮球运动 多条轨迹线	85
	习题	86

第4章 Python 编程进阶88

4.1	序列数据类型——列表	88
4.1.1	用列表存储多条轨迹线的 坐标点	88
4.1.2	列表的基本概念	89
4.1.3	列表的基本操作	90
4.1.4	用列表实现绘制多条线	92
4.2	映射数据类型——字典	93
4.2.1	用字典存储多条轨迹线的 坐标点	93
4.2.2	字典的基本概念	94
4.2.3	字典元素的访问	94
4.2.4	字典的基本操作	95
4.2.5	字典的嵌套	95
4.2.6	用字典实现绘制多条线	96
4.3	文件操作	96
4.3.1	文件的基本概念	96
4.3.2	常用文件操作	98
4.3.3	用文件存储多条轨迹线的 坐标点	102
4.4	Python 综合案例——百分之百 机器人投篮	104
4.4.1	案例任务描述	104
4.4.2	案例任务分析	104
4.4.3	案例任务实现路径	105
4.4.4	程序模块的代码实现	106
4.4.5	案例运行结果展示	109
	习题	109

第5章 算法设计112

5.1	什么是算法	112
5.1.1	算法是一种解题步骤	112
5.1.2	算法与计算思维	112

5.2	算法性能	114
5.2.1	算法性能的衡量指标：时间复杂度	114
5.2.2	算法性能分析举例	116
5.3	问题求解计算思维方法	118
5.3.1	迭代法	118
5.3.2	枚举法	119
5.3.3	二分法	120
5.3.4	递归	124
5.3.5	深度优先遍历	126
5.3.6	梯度下降法	131
	习题	134

第三部分 人工智能与智能计算

第6章 智能感知 137

6.1	自然语言处理与大模型	137
6.2	机器视觉	144
6.3	模式识别	148
6.4	案例分析与实现	151
	习题	154

第7章 机器学习 155

7.1	监督学习	155
7.2	无监督学习	157
7.3	半监督学习	159
7.4	案例分析与实现	161
	习题	162

第8章 智能决策 164

8.1	基于搜索的最优路径决策	164
-----	-------------	-----

8.2	强化学习	169
8.3	群体智能	172
8.4	案例分析与实现	176
	习题	180

第9章 智能机器人 181

9.1	机器人技术	181
9.1.1	机器人简介	181
9.1.2	机器人及其应用	182
9.1.3	机器人技术展望	183
9.2	人机交互	184
9.2.1	基于传统硬件设备的人机交互	184
9.2.2	基于触控设备的人机交互	185
9.2.3	基于智能设备的人机交互	186
9.2.4	人机交互未来发展趋势	187
	习题	190

第四部分 网络与大数据

第10章 互联网信息处理 192

10.1	网络信息获取基础	192
10.1.1	计算机网络基础	192
10.1.2	无线网络与物联网	201
10.1.3	Web 的基本工作方式	206
10.1.4	HTML 简介	208
10.2	网络爬虫与信息提取	211
10.2.1	天气数据爬取案例	211
10.2.2	网络爬虫	213
10.3	搜索引擎工作原理	216
10.3.1	搜索引擎组成	216

10.3.2	网页排名算法	218
10.4	网络安全	220
10.4.1	网络安全基本属性	220
10.4.2	OSI 网络安全体系结构	221
10.4.3	网络安全模型与网络安全访问模型	222
10.4.4	人工智能赋能的网络安全体系	223
	习题	224

第 11 章

数据管理与

大数据

11.1	计算机数据管理	226
11.1.1	人工管理阶段	226
11.1.2	文件系统阶段	227
11.1.3	数据库系统阶段	228
11.1.4	数据文件格式	229
11.2	数据库	231
11.2.1	数据库系统的组成	231
11.2.2	关系数据库	234
11.2.3	数据库设计	237
11.2.4	数据库和数据表的基本操作	240
11.2.5	人工智能赋能的数据库系统	242

11.3	大数据简介	243
11.3.1	大数据的产生	243
11.3.2	大数据的特性	244
11.3.3	大数据处理过程概述	245
11.3.4	大数据应用解决方案示例	247
	习题	249

第 12 章

数据分析

12.1	数据分析应用	251
12.1.1	数据分析定义	251
12.1.2	数据分析应用场景	251
12.2	数据分析案例	253
12.2.1	数据分析的基本步骤	253
12.2.2	气候数据分析案例	254
12.2.3	数据分析软件	257
12.3	数据分析详解	260
12.3.1	数据采集	260
12.3.2	数据处理	260
12.3.3	数据分析	261
12.3.4	数据挖掘	263
12.3.5	数据可视化	266
	习题	270

参考文献

272

第一部分

计算与计算思维

第 1 章

计算与人工智能概述

什么是计算？在数学领域，计算是抽象数学思想在具体数据上的应用，例如，用数字进行加、减、乘、除的运算。在计算机科学领域，计算是根据设定的规则将一串编码转换成另一串编码的过程。这种计算已无处不在，且广泛融入人们的生活、学习、生产和工作中。随着信息的不断发展，计算设备不仅可以安装在衣服上或放置在口袋里，还可以飞向外太空或潜入海底。人类通过思考自身的计算方式来研究计算能否由机器完成，让机器代替人类进行重复或复杂的计算，从而诞生了计算工具，发明了电子计算机，进而产生了人工智能。随着计算的日益“强大”，人工智能在很多应用领域中的作用日益突出，成为人脑的延伸。

计算思维是信息时代的一种基本的思维模式，它运用计算机科学的基础概念和计算系统进行问题求解和系统设计，深刻地影响着我们的思维方式。计算思维是计算时代的产物，是每个人都应具备的一种基本能力。

目前，人工智能不仅应用于航天、航空、探海、探月、军工、智能制造、自动驾驶等高精尖领域，也广泛应用于指纹识别、人脸识别、动作识别、专家系统、虚拟现实、扫地机器人、农业生产等生活/生产领域。可以说，人工智能的每一项应用，都是将问题转化为各类计算机能够处理的形式，最终通过计算得到结果——所有的人工智能都是应用计算思维的典范。

1.1 计算的概念

计算无处不在。作为人类，我们一直在创造计算工具以帮助我们解决问题，那么计算机到底是如何诞生的，又是如何工作的呢？

1.1.1 什么是计算

输入一个问题，按照给定的规则确定其具体的输出的过程，我们称其为计算。例如，加法问题，输入一对数值，输出是该对数值的和；又如排序问题，排序计算对每个输入表都赋一个输出表，而输出表的数据项与输入表相同，只是输出表的数据项是按照升序或降序排列的。

计算的历史十分悠久，在新石器时代，中国历史上就有结绳记事、契刻记数的记载，到春秋战国时代进一步发展出了更复杂的计算工具（如算筹）。而在大约两千五百年以前，古罗马才开始出现罗马数字。计算历史时间轴呈现了计算的起源与发展历程，如图 1.1 所示。

世界上最早的系统化计算工具是古代中国人发明的算筹，由于算筹携带不方便且容易损坏等原因，约在东汉时期（公元 1~3 世纪），中国人发明了更为方便的计算工具——算盘的雏形，后逐步发展为现代算盘并沿用至今。在 17 世纪初，英国人发明了对数计算尺，后经改进成为现代计算尺，开创了模拟计算的先河。1642 年，法国人帕斯卡（Pascal，1623—1662 年）制造出了世界上第一台机械式数字计算机（这是一种用机械技术实现数字运算的计算工具），实现了数字运算的机械化。

这些古老的计算工具虽然能够进行加、减、乘、除等运算，在当时显著地加快了人们的计算速度，但它们往往只能完成某一项特定的任务，与我们今天所理解的计算及计算机科学相去甚远。

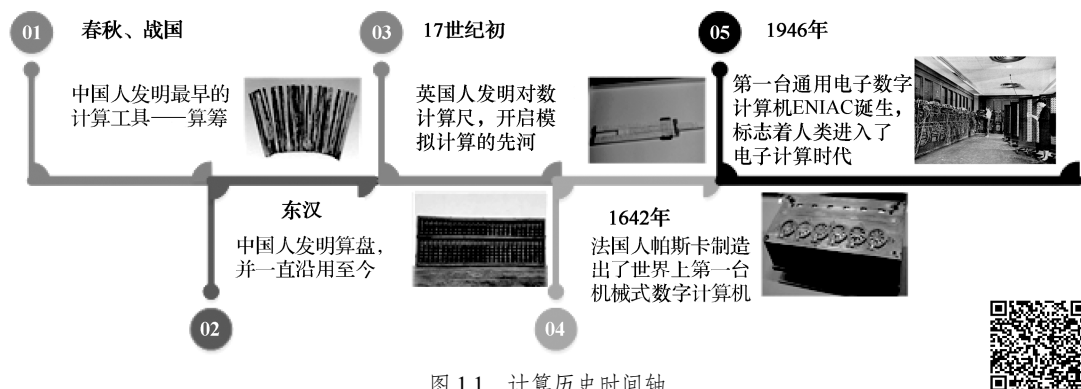


图 1.1 计算历史时间轴

在 20 世纪前半叶，随着科学技术的发展，传统的计算方式已经无法满足人们的需求。库尔特·哥德尔（Kurt Gödel），阿兰·麦席森·图灵（Alan Mathison Turing）、阿隆佐·丘齐（Alonzo Church）等一批数学家，发现当时数学界巨人戴维·希尔伯特（David Hilbert）提出的一些基本数学问题是无法求解的，确定一个数学命题是真还是假就是这样的一个例子。那么哪些问题是可计算的，哪些问题是不可计算的呢？这就是著名的可计算边界问题。为了解决这一问题，数学家们给出了研究思路：为计算建立一个数学模型，称之为计算模型，然后证明，凡是这个计算模型能够解决的问题，就是可计算的问题。

1936 年，图灵针对可计算边界问题提出了一种抽象的计算模型——图灵机（Turing Machine），将人们使用纸笔进行数学运算的过程进行抽象，由一个虚拟的机器替代人类进行数学运算。图灵将逻辑中的任意命题用图灵机来表示和计算，并按照规则推导出结论，其结果是——可计算函数等价于图灵机能计算的函数。这表明图灵机能计算的函数便是可计算的函数，图灵机无法计算的函数便是不可计算的函数。图灵机的提出证明了通用计算理论，肯定了计算机实现的可能性，同时它给出了计算机应有的主要架构，这为计算机的诞生和发展提供了理论基础。此外，图灵机模型引入了读写、算法与程序语言的概念，突破了古老计算工具的设计理念。在今天，只要在计算机上点开不同的程序，就能完成各种不同的任务。这种能解决各式各样问题的计算机都属于图灵机。

1946 年，美籍匈牙利数学家约翰·冯·诺依曼（John von Neumann）在图灵机的基础上设想了计算机的基本结构和工作方式，提出了存储程序的概念及计算机的组成和框架，奠定了现代计算机组成和工作原理的基础，最终帮助人们制造出了实用的计算机。

图灵机是理论模型，是对人计算过程的模拟，可以将其理解为现代计算机的“灵魂”，而冯·诺依曼计算机则是图灵机的工程化实现，是现代计算机的“肉体”。

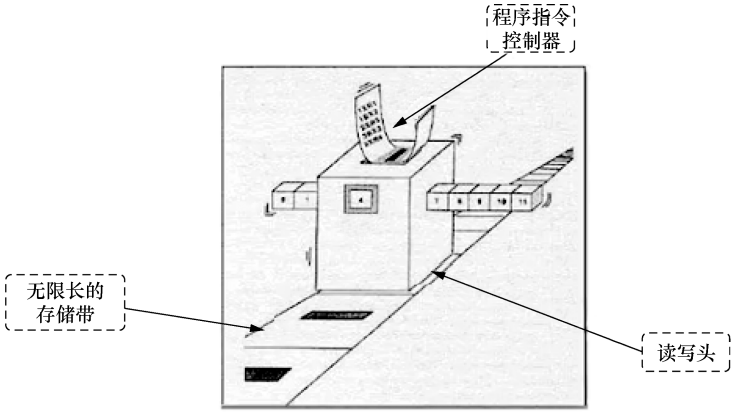
1.1.2 图灵机

在解决可计算边界问题的工作中，许多研究人员已经提出并研究了各种不同的计算模型，其中之一就是图灵机，它是由图灵于 1936 年在其发表的论文《论可计算数及其在判定问题中的应用》中提出的一种抽象计算模型。今天，图灵机仍然被用作研究算法处理能力的一种工具。

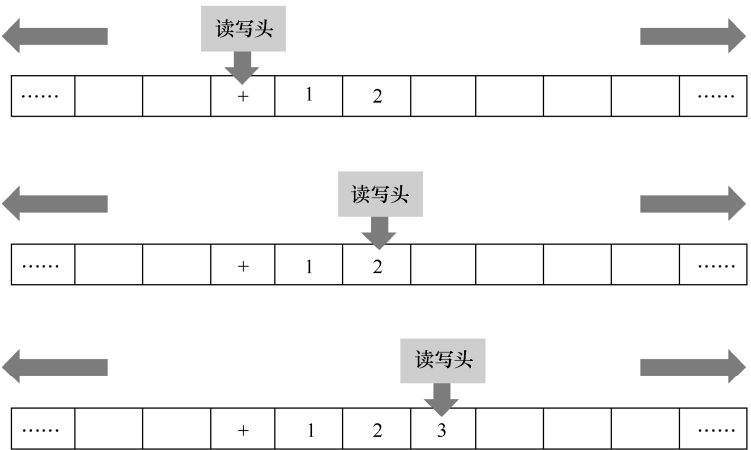
图灵机的基本原理是用机器来模拟人类用纸笔进行数学运算的过程。图灵机将一个无限长的存储带作为无限存储，它有一个读写头，能在存储带上读、写和左右移动，如图 1.2（a）所示。图灵机开始运作时，存储带上只有输入串，其他地方都是空白，如果需要保存信息，则其可以将相关信息写在存储带上。为了读取已经写下的信息，它可以将读写头来回移动到这个信息所在的位置。机器不停地计算，直到产生输出为止。

在图灵机计算的任一时刻，机器一定处在有限个条件中的一个条件下，这些条件称为状态。图灵机的计算始于一种特定的状态，称为初始状态，而停止于另一种特定的状态，称为停止状态。

接下来分析一个图灵机执行计算的具体例子：利用图灵机执行“ $1+2=3$ ”的计算。先定义读写头读到“+”之后，依次移动读写头两次并读取格子中的数据；接着读写头进行计算；最后把计算结果写入第二个数据的下一个格子里，整个过程如图 1.2（b）所示。



(a) 图灵机模型



(b) 图灵机执行“ $1+2=3$ ”的计算

图 1.2 图灵机

图灵机的结构看似十分简单，但事实上，它与算盘之类的古老计算工具有着本质的区别——它拥有一个类似于人眼和手的读写头，其能读取信息并输出信息；一条无限长的存储带，其能源源不断地提供信息或输出结果；一个类似于人类大脑的控制器，其能根据不同的问题进行不同的处理。这意味着，图灵机有无限大容量的存储和可以任意访问的内部数据，在控制器中输入不同的程序，它能够处理不同的任务。图灵机实际上是一种精确的通用计算机模型，能模拟实际计算机的所有计算行为。图灵机模型是应用非常广泛的经典计算模型，为现代计算机提供了理论原型。

1.1.3 什么是计算机

智能移动机器人的路径规划中求最短路径是一个有趣的经典问题，在没有任何先验知识的情况下，机器人只能在尝试所有路径之后，找出其中的成功路径，再比较所有成功路径，进而确定最短

路径。显而易见，这是一件非常耗时的事情，而我们可以借助计算机来求答案。

计算机作为提高效率的工具，它是如何诞生的，又是如何工作的呢？本小节将从路径规划过程中求最短路径问题开始，设计程序来求解这个问题，并分析计算机如何进行计算，程序如何编写，以及在计算的过程中计算思维是如何发挥作用的。

1. 冯·诺依曼体系结构

1946 年，冯·诺依曼在图灵机模型的基础上提出了奠定现代计算机基础的冯·诺依曼结构，该结构指出了计算机体系结构的 3 个基本原则：计算机采用二进制逻辑、计算机遵循存储程序原理、计算机由 5 个部分组成（运算器、控制器、存储器、输入设备和输出设备），如图 1.3 所示。冯·诺依曼体系结构是现代计算机的标准结构。这种结构的特点是“程序存储，共享数据，顺序执行”，需要中央处理器从存储器取出指令和数据进行相应的计算。

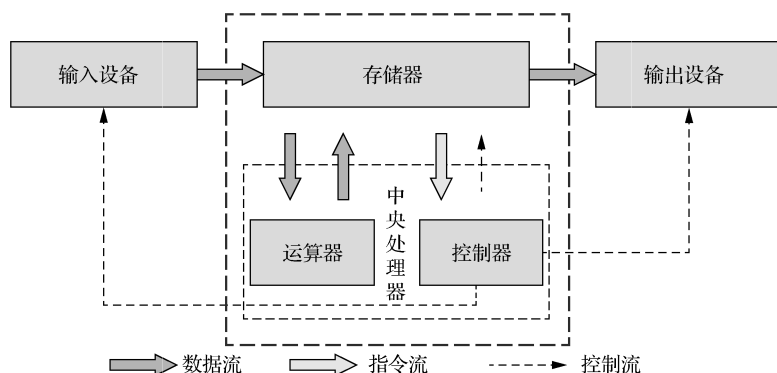


图 1.3 冯·诺依曼体系结构

求解最短路径，从冯·诺伊曼体系结构的角度看是如何进行的？首先，从传感器采集或从键盘输入程序数据，不同的数据进入计算机系统后都会被保存在存储器里。然后，由操作系统控制中央处理器进行运算和存储。计算机从存储器里取出数据并传送给中央处理器，其中的运算器和控制器将根据算法来操作和改变这些数据，一系列的命令处理完成后重回存储器。最后，这个过程反复递归直到输出结果，并在输出设备上显示程序执行所得的结果。

2. 计算机系统

基于冯·诺伊曼体系结构的计算机系统如图 1.4 所示，主要包括计算机硬件和计算机软件，硬件包括中央处理器（运算器、控制器）、存储器和输入输出等设备；软件分为系统软件和应用软件两大类。

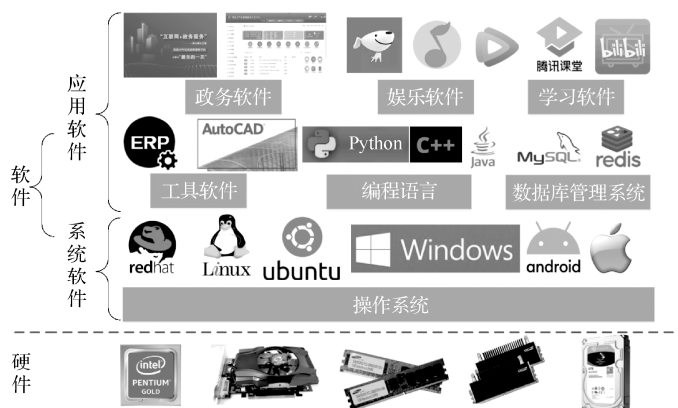


图 1.4 计算机系统

1946 年 2 月 15 日发生了人类历史上一件划时代的大事：世界上第一台通用电子数字计算机（见图 1.5）——ENIAC（埃尼阿克）计算机诞生了，它实现了计算工具由机械计算机到电子计算机的过渡，在计算机发展史上具有划时代的意义。以圆周率的计算为例，中国古代科学家祖冲之利用算筹，耗费 15 年心血，才把圆周率计算到小数点后 7 位数；一千多年后，英国数学家香克斯（Shanks）以毕生精力计算圆周率，才计算到小数点后 707 位数；而使用 ENIAC 计算机进行计算，仅用 40s 就达到了这个记录，还发现在香克斯的计算中，第 528 位是错误的。



图 1.5 世界上第一台通用电子数字计算机

埃尼阿克计算机最初的设计方案由莫奇利（Mauchly）于 1943 年提出。它采用 17 468 个电子管，工作耗电 147kW，质量 30t，占地面积 167m²，可谓庞然大物。虽然第一台计算机存在许多缺点，但是它显著提高了计算速度，这在人类计算工具的发展史上无疑是一个巨大的飞跃。

科学技术在不断发展，计算机芯片也在不断地推陈出新，从早期的继电器、电子管发展为晶体管，直到今天大规模使用的集成电路。20 世纪 50 年代中后期，使电子器件和设备更加小型化和可靠、价格更为低廉，已成为迫切的需求。1959 年集成电路诞生，1971 年以特德·霍夫（Ted Hoff）为首的研究小组完成了世界上第一款微处理器芯片 Intel 4004（见图 1.6）的制造，他们在 3mm×4mm 的面积上集成了 2 300 个晶体管（10μm 工艺），时钟频率 740kHz，每秒运算速度约 9 万次。这意味着中央处理器已经微缩成了一块集成电路，芯片上的计算机由此诞生。

今天，计算机已无处不在，其应用领域如图 1.7 所示。计算机系统不断发展，逐步出现了各行业、各领域的专业计算机，以及面向家庭、个人、社会和政府的生活/工作计算机等。它可以是帮助人们进行各种计算的工具，是各种机器的大脑，能以看不见、摸不着的形式出现并为人类服务；也可以是人们跨越时空进行交流的工具，或是改造自然的工具；还可以是人类创造另一个虚拟世界的工具。

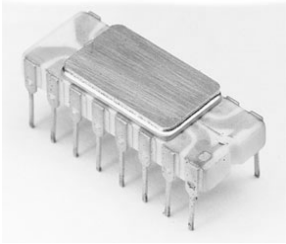


图 1.6 第一款微处理器芯片 Intel 4004

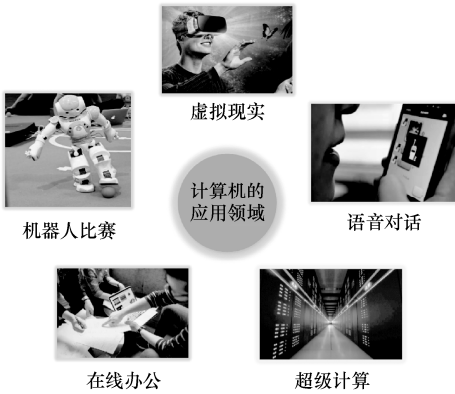


图 1.7 计算机的应用领域

3. 计算机硬件

计算机的硬件部分一般包括中央处理器、存储器（内存、硬盘）、输入设备（键盘、鼠标）、输出设备（显示器、音箱）。输入设备能够指挥计算机做事情。用户可以通过输入设备（如键盘、鼠标、麦克风、摄像头等）告诉处理器芯片要做什么。如今，输入设备的表现形式越来越多，比如把计算机嵌入手环里，它能听到心跳；把计算机嵌入汽车里，它能知道汽车的运行状态；触摸屏能感受到手指的滑动并接收输入信息等。计算机输出信息的方式也有很多种，比如，计算机显示器可以显示文本、照片、视频，音箱可以输出声音。计算机还可以借助多种输出设备来虚拟现实；计算机的输出设备还包括控制机器人行动等的功能设备。

计算机的工作离不开硬件的支持。例如，在探索最短路径的过程中，首先需要从键盘输入机器人所处环境的地图，然后由操作系统将地图数据传送到内存，由中央处理器对数据进行运算并获得结果，最后在显示器上输出寻找到的最短路径。

中央处理器是计算机的核心硬件，它往往以处理器芯片的形式存在。在现实生活中，各类处理器芯片无处不在，多样化的处理器芯片控制着各种机械设备运行。从图 1.8 可以看出，交通中的汽车、平衡车等，日常生活中的冰箱、洗衣机等智能家居设备，都与处理器芯片紧密相关。道路上行驶的汽车内安装有多种传感器、控制器并受处理器芯片控制，一辆车有 70 种以上的电子控制单元（Electronic Control Unit, ECU）芯片，越高档的汽车，芯片的数量就越多，辅助驾驶、无人驾驶车辆内含有的电子产品更多。



图 1.8 处理器芯片控制的各类设备

4. 计算机软件

硬件有形，而软件则无形。计算机软件多种多样，如图 1.9 所示，它可以是计算机系统上的操作系统、应用软件，也可以是用户与硬件之间的接口（驱动）程序，还可以是运行在硬件上的各种算法程序和用户程序。在求解最短路径问题的过程中，操作系统是系统软件，它通过内存管理（如虚拟内存）进程调度（如 CPU 时间片分配）和文件系统等机制。实现对硬件资源的抽象管理。人们利用程序语言编写不同的算法来实现不同的功能，算法是有限步骤的问题求解方法，不同算法的设计直接决定了程序运行结果的正确性和效率，而程序是算法的具体实现。使用不同的算法来求解最短路径，其时间复杂度差异显著。



图 1.9 计算机软件

2011 年 8 月 20 日，美国企业家马克·安德森（Marc Andreessen）在《华尔街日报》专栏《为何软件正在“吃掉”全世界》（“Why Software is Eating the World”）中刊登文章称，软件正在占领全世界，他认为全球的各行各业正在向以软件为基础的方向转变，越来越多的大企业和行业开始依靠软件运行并提供在线服务。他预言在十年后会有更多的行业被软件瓦解。根据互联网数据中心（International Data Corporation, IDC），2023 年全球企业软件支出达 7 560 亿美元，较 2011 年的 2 940 亿美元增长 157%，从而有效印证了这一发展趋势。

1.1.4 算法、计算机语言与程序

实现计算的自动化需要解决数据的数字化表示、机器的自动存储和程序的自动执行等问题。计算的自动化需要依靠计算机语言、程序和算法。算法是计算系统中程序的灵魂，是计算机求解问题的步骤序列的表达。计算机语言包括书写规范、语法规则、标准集合等，是人和机器都能理解的语言，它将算法描述成计算机可执行的程序指令。程序是一系列计算规则的表达，是计算机能够理解的解决问题的步骤。算法要通过计算机语言来实现，用计算机语言将算法编写成程序之后，计算机就能够理解解决问题的步骤并执行相应的程序。算法、计算机语言和程序之间的关系如图 1.10 所示。

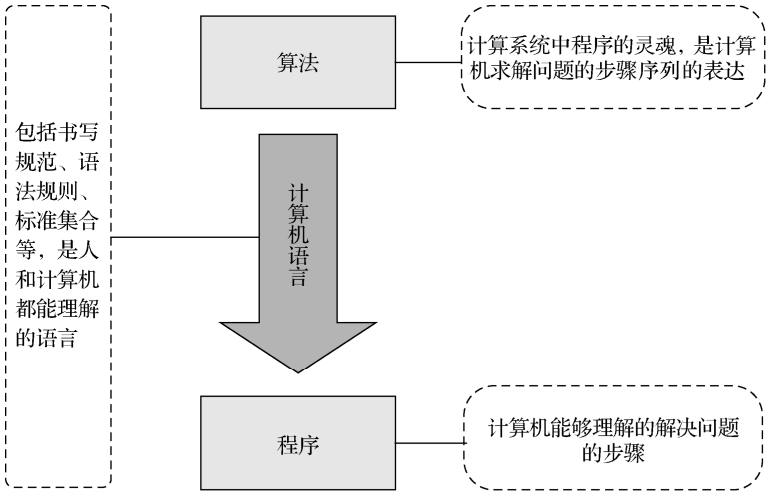


图 1.10 算法、计算机语言与程序之间的关系

为了找到智能移动机器人行走的最短路径，可以先通过算法来设计计算的步骤，再通过计算机语言将算法描述成程序，这样就可以通过计算机来执行所设计的程序。在手动进行计算的时候，我们需要知道具体的计算规则，规则可能很复杂，但是计算量不能太大，否则需要花费太多的时间；而计算机是自动计算的，只要将规则设计好，它就可以完成计算量很大的工作，这正是计算机（作为一种计算工具）计算能力的体现。

求智能机器人在环境中行走的最短路径时，需要重复不断地求出当前位置与下一个位置之间的距离。假设已知机器人已经行走的距离为 2cm，机器人还需要往前走 4cm，那么机器人总共需要行走的路径长度为多少呢？这其实是一个求和问题，接下来以这个求和问题为例，说明高级语言源代码及其翻译成指令（即编译）的过程。如图 1.11 所示，要求解求和问题，首先分解步骤，定义变量，对两个距离进行求和并存储结果，这就是求和的算法；然后，通过高级程序的语言将它程序化，再通过编译系统将编写好的程序变成计算机能够读懂的指令；最后，计算机实现指令数据的存储和执行，执行完毕将结果返回给用户。

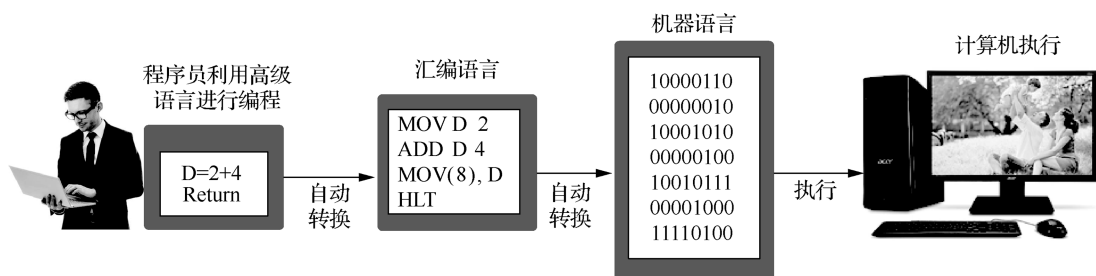


图 1.11 高级语言源代码及其编译过程示意

1. 计算机语言的发展

计算机语言与自然语言不同，它是可以通过确定性的方法被计算机解析、编译或解释的一种形式语言。它有基本的语法规则。计算机语言一般分为机器语言、汇编语言和高级语言三大类，如图 1.12 所示。

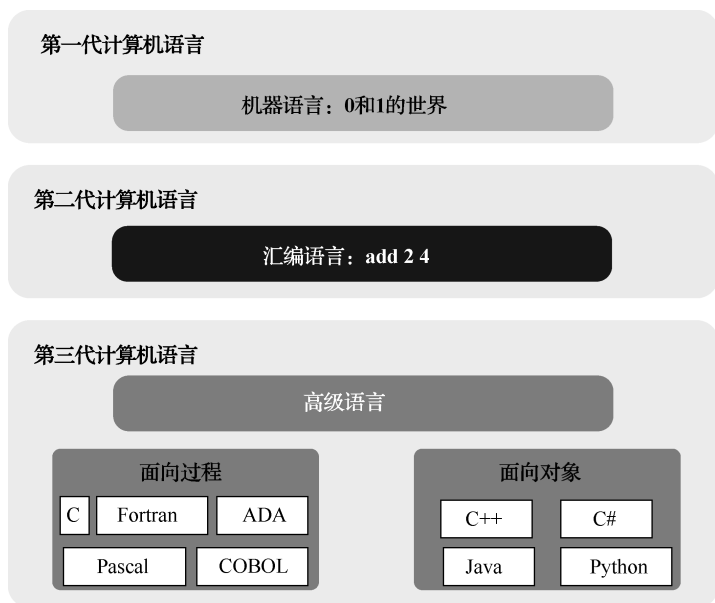


图 1.12 计算机语言分类

第一代计算机语言是机器语言，具有二进制编码形式，由其编写的代码是由 0 和 1 组成的计算机能够直接识别和执行的指令码。尽管机器语言对于计算机的工作来说是快速和直接的，但是能使用机器语言的人非常少，因为机器语言难以理解，可读性差。例如，图 1.11 给出的某台计算机上的一串二进制指令编码：10000110 00000010 10001010 00000100 10010111 00001000 11110100。其含义是先将 2 赋给 D，然后计算 D+4 的值并赋给 D，最后将 D 的内容存储到地址为 8 的存储单元中。这一串二进制指令的目的是计算 2 和 4 相加的和。从这个例子可以看出，简单的加法求和问题，机器语言需要使用多串二进制指令配合求解，而对于编程人员来说，记住每一串由 0 和 1 组成的指令是一件十分困难的事情。因此，使用机器语言编写程序是非常不方便的。

为了解决机器语言面临的问题，第二代计算机语言（即汇编语言）诞生了。汇编语言用英文字母或者符号串来替代机器语言，把不容易理解和记忆的 0 和 1 的代码对应关系转换成汇编指令。比如说，在“add 2 4”这条汇编指令中，add 是相加的英文单词，指令表示 2 和 4 相加求和。汇编语言比机器语言更加便于阅读和理解，但是汇编语言依赖于硬件，这使得程序的可移植性极差。

尤其是编程人员在使用新的计算机的时候，还需要学习新的汇编指令，这极大地增加了编程人员的工作量。

为了解决汇编语言存在的问题，第三代计算机语言——高级语言就诞生了。高级语言不是一门语言，而是一类语言。它比汇编语言更接近于人类使用的语言，比较容易理解、记忆和使用。例如，C、C++、Java、Python 等都是高级语言，它们和计算机的架构指令集无关，一般具有较好的可移植性。

2. 问题求解策略与算法

无论使用什么样的语言，算法都是计算的核心。算法描述的是计算机求解问题的步骤序列，而程序能否编写成功，重要的是看能否找出问题求解的算法。以求解机器人行走的最短路径为例，利用遍历算法和贪心算法分别求解，并比较两种算法的效率。

针对最短路径求解问题，我们可以将其抽象为一个组合优化问题，即：机器人从起点走到终点的过程中，有 n 个节点需要访问，按怎样的次序访问，才能使机器人访问路径的“权值和”最小？最常使用的求解最短路径问题的方法是遍历算法和贪心算法。

遍历算法思路比较简单，所谓“遍历”，就是将问题的每一种可能解代入问题中进行计算，通过对所有可能解的计算结果的比较，选取能满足约束条件和目标的解作为问题的最终解。利用遍历算法求解最短路径时，可以把复杂的路径分解求解，把可能短的每一条路径通过枚举法列举出来，再比较得出最短路径，该最短路径即为问题的解。遍历算法的流程图如图 1.13 所示。

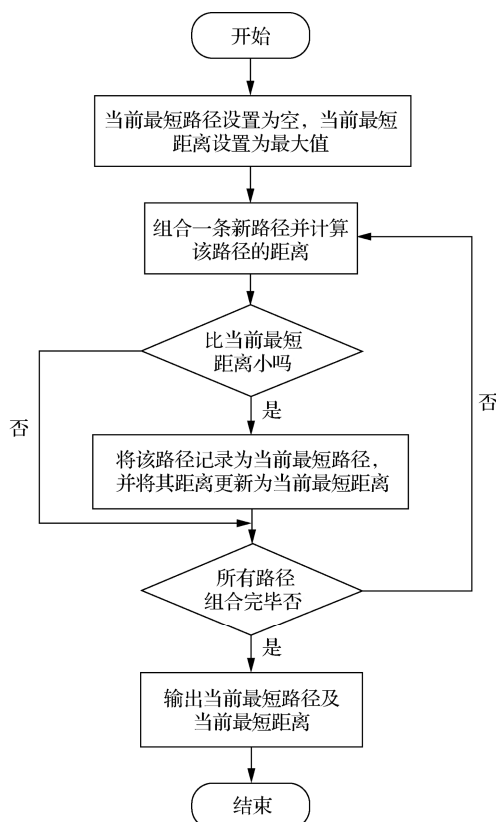


图 1.13 遍历算法求解最短路径流程图

遍历算法把一个复杂的庞大的计算过程转换为简单过程的多次重复，这种算法利用了计算机计算速度快而不知疲倦的特点。然而，遍历算法存在一个问题，即能否在有限时间内求解的问题，这

与问题的规模有关。这里所说的规模一般是指问题求解计算量的影响因素，例如，最短路径问题中满足约束条件的解，即 n 个节点的所有组合所形成的路径，其数目是 $(n-1)!$ ，因此其时间复杂度为 $O(n!)$ 。随着地图节点数目的不断增大，组合路径数将呈指数级规律急剧增长，以致达到无法计算的地步。对于这类大规模难解问题，需要用其他快速的方法来求解。因此，寻找时间上切实可行的简化求解的方法就成为了问题的关键。

利用贪心算法可以改进求解大规模最短路径问题算法的性能。贪心算法是一种局部优化的求解算法，其核心思想是一定要在当前情况下做最佳选择，以防止将来后悔，故名“贪心”。机器人从某一个节点开始，每次选择一个节点，直到所有节点都被选完。机器人每次在选择下一个节点时，只考虑当前情况下的最佳选择，保证迄今为止经过的路径总距离最短，即在其所有能到达的节点中选择距离最短的那个节点作为下一个被选中要经过的节点，该算法的流程图如图 1.14 所示。当最短路径问题规模过大时，只能退而求其次，使用贪心算法求近似最优解，以此来降低时间复杂度。贪心算法牺牲了结果的精度（即求得的不一定是最优解），但换来了时间上可观的节约，如将算法复杂度直接降到多项式量级 $O(n^3)$ 。

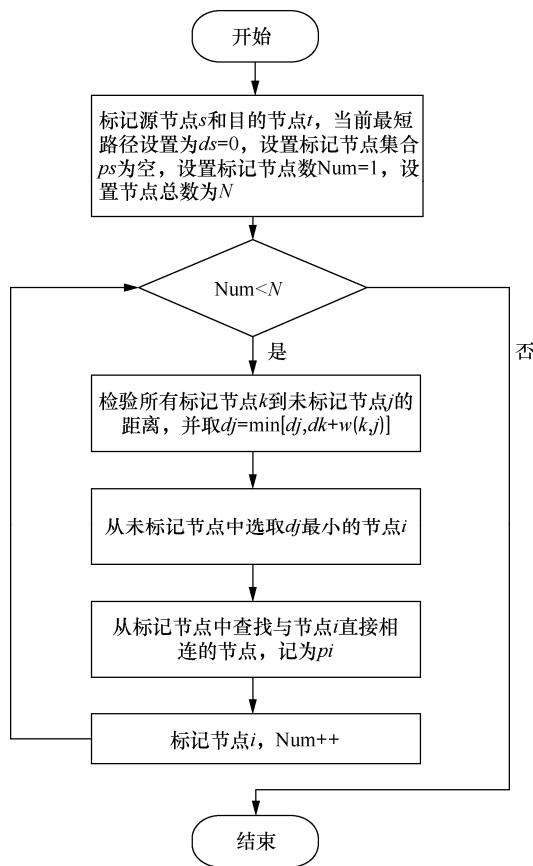


图 1.14 贪心算法求解最短路径流程图

怎样来判断一个算法的好坏呢？可以通过分析时间和空间的复杂度来实现，尤其需要关注在最坏情况下它的复杂度是多少。对于最短路径问题，当问题规模小时，可以直接枚举问题空间得出最优解；当问题规模大时，一般只能靠近似算法求出近似最优解。而算法的性能和机器的硬件密切相关，运算能力越强的机器越可以提高最终的执行效率。

第三部分

人工智能与智能计算

第6章

智能感知

人在感知周围环境时依赖于自身的视觉、听觉、触觉等自然感知能力，而当机器像人一样具备理解语言、识别图像、聆听声音等感知能力时，机器就有了认知世界和响应世界的可能。机器感知是人工智能的重要内涵之一。随着深度学习的发展，自然语言处理、机器视觉、模式识别等作为人工智能在机器感知层面的重要研究内容，已经在许多领域得到了广泛应用。

6.1 自然语言处理与大模型



20 世纪中期第一台通用电子数字计算机问世，人们便产生了机器可以准确翻译自然语言的美好愿景，并希望可以创造出一台机器，实现各种语言之间的互通和转换。人们开始思考：计算机是否可以处理自然语言？计算机处理自然语言的方式是否和人类一样？人类理解自然语言的过程如图 6.1 所示。

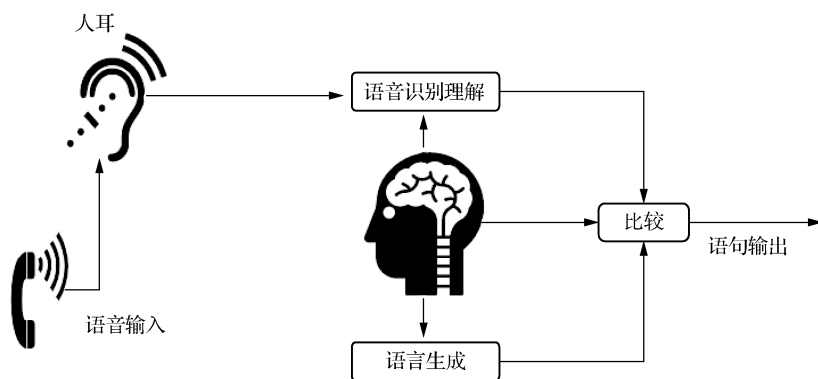


图 6.1 人类理解自然语言的过程

参照人的翻译过程，两种语言之间的转换可以简单地使用“逐词替换”来实现，例如，想把英语语句翻译成汉语语句，只须先将句子分解为单词，再用对应的汉语单词进行替换，最后按照汉语的语法规则整理语句顺序：

This is a computer
这 是 一（台） 计算机

这种逐词替换的方法需要大量的词汇存储及快速的搜索速度来保证两种语言快速对应。但是，逐词翻译引出了自然语言处理中的经典难题——歧义问题。假设我们将某个语句输入计算机，但是这个语句的含义不止一种，例如：

I made her goose

“goose”和“her”在句法和语义上都是有歧义的。“her”可以表示给予格的代词或者所属格的代词，“goose”既可以是名词，也可以是形容词。如图 6.2 所示，“翻译软件 1”将“goose”视为名词“鹅”，并按双宾语结构对该语句进行处理，将该语句译为“我给她做了鹅”；但是，“翻译软件 2”将“goose”视为形容词，并按宾语补足语结构对该语句进行处理，将该语句译为“我把她惹毛了”。



图 6.2 机器翻译的不同结果

那么怎么让机器真正理解并生成人类的自然语言呢？自然语言处理就是致力于此的一项技术。

1. 自然语言处理的方法

在自然语言处理的发展过程中，产生了基于规则的自然语言处理方法和基于统计的自然语言处理方法。

基于规则的自然语言处理方法是理性主义的，它这样处理自然语言：依照人们所设定的自然语言语法规则将输入的语句分解为句法结构，如图 6.3 所示，再根据一套语义规则把句法结构映射到语义符号结构。在这套自然语言处理系统中，规则集合是人们预先设计给机器的，是先验的知识。

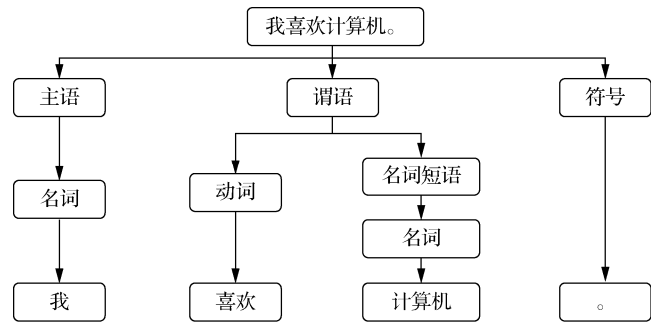


图 6.3 句法结构分析树

理性主义在 20 世纪 50 年代至 20 世纪 80 年代盛行，它认为人类的大部分语言知识是与生俱来的，这些语言知识构成了分析和理解自然语言的规则。在一个自然语言处理系统中，与领域相关的规则集由专家手工编写，因此理性主义受限于人类学习语言的方式和语言研究的深度。

除了句法，研究者也关注语义分析和知识表示等问题，但是语义比句法更难通过计算机描述和表达。值得一提的是，中国古代语言学的主要研究集中在语义而不是句法上，例如，《说文解字》就是对语义的研究成果。

图 6.4 所示，早期人类对自然语言处理的理解认为其是基于句法分析和语义分析的。

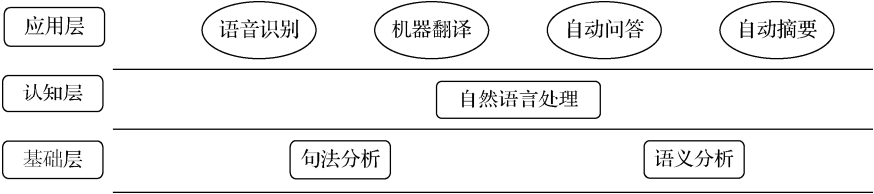


图 6.4 早期人类对自然语言处理的理解

当人们意识到规则无法灵活地应对语言问题时，基于统计的经验主义开始受到重视。经验主义认为人类不可能天生就拥有一套语言的处理规则和方法。经验主义尝试从大量的语言数据中获取语言的结构知识，这开辟了基于语料库的研究方法。经验主义利用统计学习的方法建立语言处理模型，并由语料库中的训练数据来估计统计模型中的参数。

表 6.1 对理性主义与经验主义进行对比。

表 6.1 理性主义与经验主义对比

理性主义	经验主义
继承哲学中的理性主义	继承哲学中的经验主义
以符号为主，认为人类的行为可以用物理符号系统进行模拟	使用随机或者概率的方法研究语言
演绎法	归纳法
有限状态转移网络，有限状态转录机合一算法，依存算法，一阶谓词演算，语义网络	隐马尔可夫模型，最大熵模型， n 元语法，概率上下文无关语法，噪声信道理论，贝叶斯方法

基于统计的经验主义在文字识别领域因大量的资料分析而取得了巨大的成功。20 世纪 50 年代末到 60 年代中期，自然语言处理的经验主义蓬勃发展。1959 年，威廉·W·布莱德索（William W.Bledsoe）和罗伯特·布罗宁（Robert Browning）建立了贝叶斯系统以用于文本识别；1961 年世界上第一个联机语料库——布朗语料库诞生了。20 世纪 60 年代，统计方法在语音识别上也取得了巨大的成功，具有代表性的成果有美国数学家伦纳德·E·伯姆（Leonard E.Baum）等人提出的隐马尔可夫模型（Hidden Markov Model, HMM）和卡内基-梅隆大学（Carnegie Mellon University, CMU）的詹姆斯·K·拜克（James K.Baker）等人提出的噪声信道模型。

2. 自然语言处理的任务

自然语言处理有两个核心的任务，一个是自然语言理解（Natural Language Understanding, NLU），另一个是自然语言生成（Natural Language Generation, NLG）。

自然语言理解，顾名思义就是计算机怎样理解自然语言，以及具备和正常人一样的语言理解能力，其核心问题：自然语言是如何传递信息的？人如何理解语义并掌握真正的信息？自然语言理解在生活中的应用无处不在，如机器翻译、语音助手、聊天机器人，目前其技术已经达到可以使计算机、手机等电子设备快速理解人类的意图的水平。

自然语言生成涉及自然语言处理的很多领域，如翻译、写新闻稿、诗词创作、视觉问答等。自然语言生成就是将计算机语言转换成人能够理解的自然语言，如图 6.5 所示。计算机依靠知识库或逻辑形式将计算机内部的表达形式转换为人能理解的自然语言，这可以理解为自然语言理解的逆向过程。

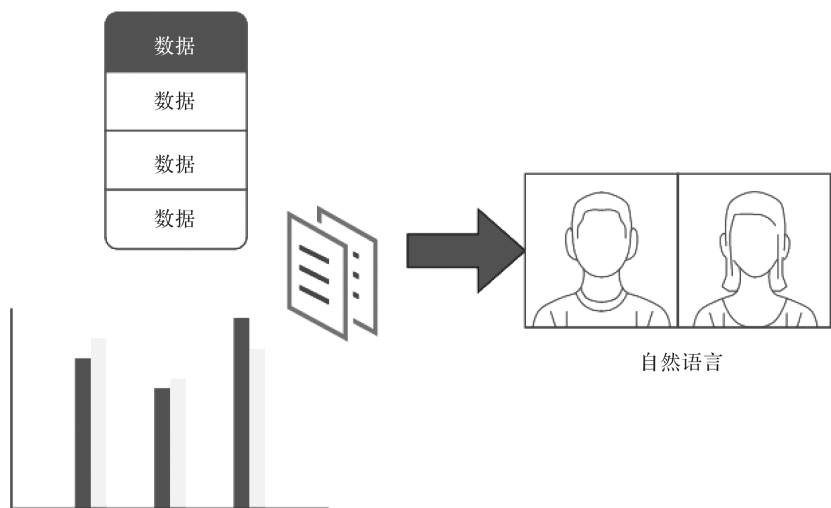


图 6.5 自然语言生成

以对话系统中的自然语言生成为例，对话系统按功能划分为闲聊型、任务型、知识问答型和推荐型。在不同类型的对话系统中采取的自然语言生成方式也不相同。闲聊型对话系统根据上下文对提问者进行意图识别和情感分析，然后生成开放性回复。知识问答型对话系统对问句进行识别与分类，然后通过信息检索或文本匹配生成用户需要的知识，如图 6.6 所示。



图 6.6 知识问答型对话系统自然语言生成实例

3. 自然语言处理的应用

自然语言处理早已深入我们的生活，为日常出行提供便利。输入法的自然联想功能就是自然语言处理的一个典型应用。十几年前人们普遍喜欢使用五笔字型输入法，就是因为当时的拼音输入法没有自然联想功能，输入速度慢。直到统计语言模型可以根据一长串的拼音序列自动选择最有可能的汉字序列，拼音输入法才逐渐取代了五笔字型输入法。

如图 6.7 所示，机器翻译和语音助手都是生活中的自然语言处理应用，它们为人们的日常生活和学习提供了便利。机器翻译是自动将一种语言转换为另外一种语言并保持语义完整的处理过程，是自然语言处理的一个分类。在早期，机器翻译依靠规则和字典，效率并不高，引入神经网络后，庞大的数据提供了极高的可用性，加上功能强大的计算机，机器翻译结果已经十分准确了。

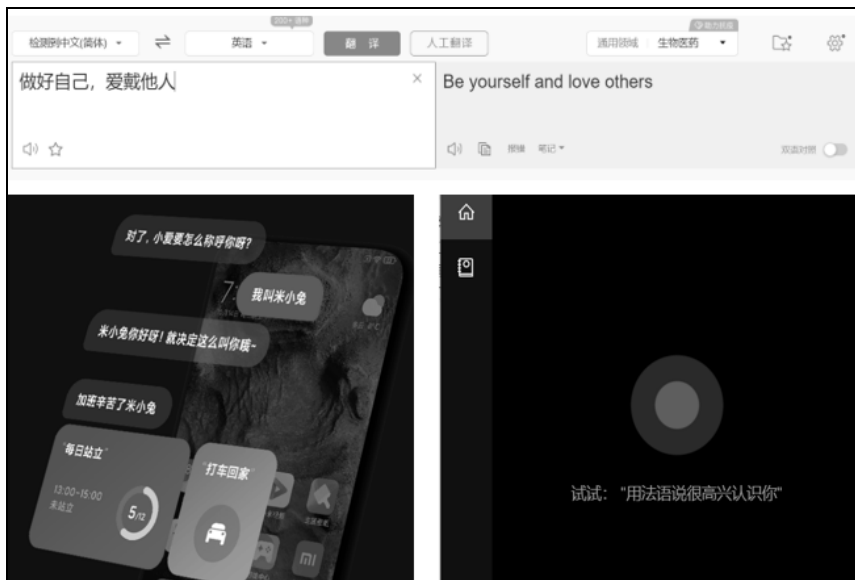


图 6.7 生活中的自然语言处理应用

语音助手可以对我们说的话进行语音识别并分析语义, 从而对我们提出的问题进行回答。这类应用已经成为我们日常生活中极其重要的一部分。而我们在使用搜索引擎搜索信息时, 往往只需要打出几个字就能得到需要的信息, 如果输入了错误的字母或符号, 机器也会细心地纠正并且帮助我们搜索到正确的信息。

4. 大模型及其应用

(1) 什么是大模型

近年来, 随着深度学习技术的不断发展, 大模型 (Large Model) 在自然语言处理领域崭露头角。大模型, 顾名思义, 指的是参数规模巨大、结构复杂的人工智能模型。这些模型通常基于深度学习框架, 通过海量数据进行训练, 能够捕捉数据中的复杂模式和深层关系。与传统的小模型相比, 大模型在处理能力、泛化能力和智能化水平上有着显著的优势, 能够处理更复杂的任务。

大模型的发展始于 20 世纪 90 年代, 经历了从传统神经网络到深度学习的转变。2017 年, Transformer (变换) 架构的提出标志着大模型时代的开启。2018 年, 基于 Transformer 的双向编码器表征 (Bidirectional Encoder Representations from Transformers, BERT) 模型的发布推动了自然语言处理领域的大模型发展。2020 年, 生成式预训练 Transformer (Generative Pre-trained Transformer, GPT)-3 的发布将大模型参数量推向千亿级。2022 年, 生成式预训练变换模型 (Chat Generative Pre-trained Transformer, ChatGPT) 的火爆进一步引发了人们对大模型的关注。近年来, 大模型在多模态融合、参数规模、应用场景等方面不断突破, 成为人工智能领域的重要力量。

(2) 大模型的基本原理

大模型通过大量标注或未标注的数据进行训练, 利用 Transformer 深度学习算法自动学习数据中的特征表示。这些特征表示能够捕捉到数据的内在结构和规律, 使得模型在未见过的数据上也能表现出良好的泛化功能。

Transformer 模型最初设计用于自然语言处理任务, 其关键核心技术是自注意力 (Self-Attention) 机制。注意力机制是一种模仿人类认知系统中注意力分配行为的技术, 旨在让神经网络在处理输入数据时能够选择性地关注相关信息, 而自注意力机制则进一步让模型在处理单个序列时考虑该序列内部各元素之间相互关系的能力。因此, 自注意力机制使得模型不仅关注当前处理的元素本身, 还能够综合考虑其他相关元素的影响, 以此增强对序列整体的理解。

大模型的训练分为三个主要阶段，包括预训练（Pretraining）、监督微调（Supervised Finetuning, SFT）和人类反馈学习（Reinforcement Learning from Human Feedback, RLHF），如图 6.8 所示。预训练阶段是大模型训练的第一步，通过给模型“喂食”海量的文本数据，让模型自主学习（无监督学习）并发现数据中的规律和模式，建立对语言或其他数据的初步认识和理解，这个阶段通常会消耗 99% 的计算资源。预训练完成后，模型进入监督微调阶段，这个阶段需要人工介入，提供高质量的标注数据，对模型进行有针对性的训练（监督学习）。通过这个阶段，模型能够针对特定任务进行调整，提升在特定任务上的表现能力，例如，文本分类、机器翻译等。最后是人类反馈学习阶段，在这个阶段，模型在实际应用中会与用户进行交互，收集用户的反馈，通过这些反馈，模型不断调整和优化自身的参数和策略，提升回答问题的质量和用户的满意度，这个过程通常是持续不断的，模型会从用户的反馈中持续学习，逐渐改进自己的回答策略和质量。

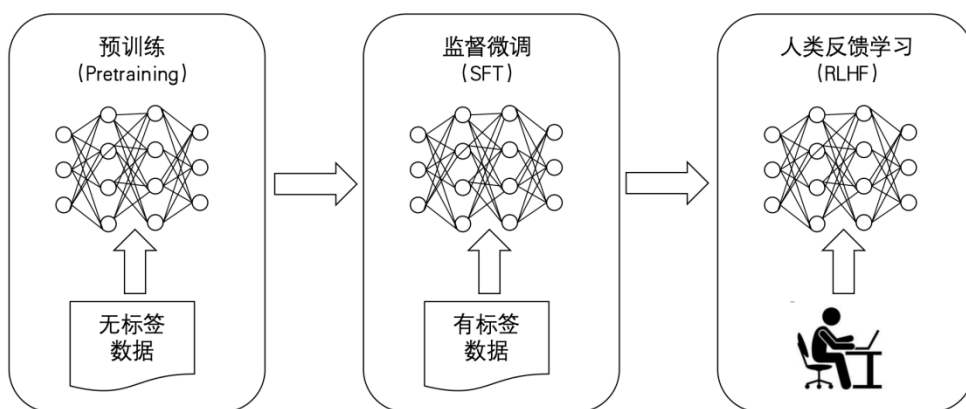


图 6.8 大模型训练的三个阶段

（3）典型大模型介绍

截至 2024 年年底，全球共发布了一千多个大模型，其中国内共有 244 个大模型发布，这些大模型在自然语言处理、计算机视觉等多个领域都有广泛应用，如 OpenAI 的 ChatGPT、华为的盘古大模型、百度的文心一言、阿里巴巴的通义千问，以及深度求索的 DeepSeek 等。

ChatGPT 是由 OpenAI 开发的基于 GPT 技术的人工智能模型，以其强大的自然语言处理能力著称。它能够模拟人类的对话，回答各种问题，并支持多语种、多领域的应用。ChatGPT 具备多功能性，可以完成撰写邮件、视频脚本、文案、翻译、代码等多种任务。其训练数据主要来源于英文世界的对话、文章等文本资源，因此在处理英文文本时具有极高的准确性和流畅性。由于训练数据以英文为主，其在处理中文文本时可能存在一定的局限性。

盘古大模型是华为旗下的超大规模 AI 大模型，涵盖了自然语言处理（Natural Language Processing, NLP）、计算机视觉（Computer Vision, CV）和科学计算等多个领域。盘古大模型具有多模态能力，能够理解和处理包括文本、图片、视频等多种模态的信息。其技术特点包括端侧至云端全系列模型、强思维能力、数据合成技术等。盘古大模型在政务与公共服务、工业与制造业、气象预测、医疗健康等多个领域都有广泛应用。

百度的文心一言是基于文心大模型技术推出的生成式对话产品，更侧重于中文文本的生成。文心一言在文学创作、商业文案创作、数理逻辑推算、中文理解和多模态生成等方面具有显著的优势。它能够从海量数据和大规模知识中融合学习，具有丰富的知识储备，并通过检索技术快速找到相关信息，为用户提供更高效的答案。文心一言的对话能力自然流畅，能够更好地满足用户的个性化需求。与 ChatGPT 相比，文心一言在处理中文文本时更具优势。

DeepSeek 是深度求索公司推出的新一代大模型，专注于高效、精准的自然语言处理和多模态任务。其核心优势在于高效推理与低资源消耗、多模态能力、中文优化和行业定制化。DeepSeek 通过创新的模型压缩和加速技术，能够在保证性能的同时显著降低计算资源需求，适合在边缘设备和移动端部署，同时，其支持文本、图像、音频等多种模态的输入与输出，能够实现跨模态的信息理解与生成。

（4）大模型的提示词

提示词（Prompt）是引导模型生成特定内容或回答特定问题的关键词或短语，通过输入相关的关键词或短语，用户可以精准地控制大模型的输出，提高对话的针对性和效率，满足个性化的需求和场景应用。以文心一言为例，撰写提示词可以遵循如下规则。

① 明确任务类型。提示词的基础公式：Prompt=任务+生成主体+细节（可选）+形式（可选）。例如，“请生成一篇关于人工智能发展的报告”，其中“生成一篇报告”是任务类型，“人工智能发展”是生成主体。

② 提供具体细节。为了获得更具体、更有针对性的输出，可以在提示词中提供尽可能多的细节。例如，“请生成一篇关于人工智能在教育领域应用的深度分析文章，要求包含至少三个实际案例”。

③ 优化提示词结构。图 6.9 所示，使用结构化的提示词可以帮助模型更好地理解我们的需求。例如，“#任务#：撰写一篇新闻稿。#主题#：嫦娥六号探测器成功登月。#要求#：内容详实，语言流畅。#受众#：科技媒体。#字数#：300”。

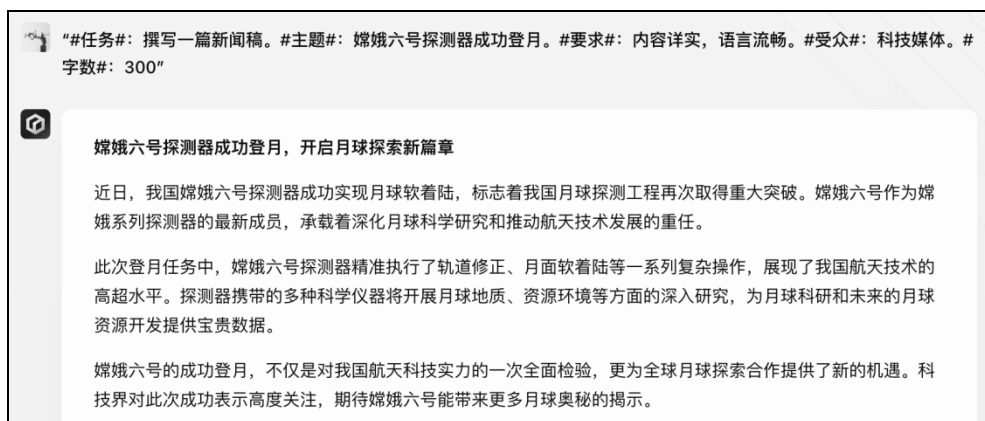


图 6.9 使用结构化提示词

④ 避免歧义和模糊性。确保提示词中的每个词汇和短语都清晰、明确，避免使用可能引起歧义的词汇或表述方式。

⑤ 尝试多种提示词。如果第一次的提示词没有得到满意的结果，可以尝试调整提示词的表述方式或添加更多细节，以便模型更准确地理解我们的需求。

（5）大模型的编程接口

大模型的编程接口（Application Programming Interface, API）是连接开发者和预训练大模型的桥梁，允许开发者通过编程调用大模型进行各种任务。这些接口支持文本生成、理解、翻译、问答等多种功能。开发者无须自建模型，即可利用 API 快速构建应用，如智能客服、内容创作辅助等。通过 API，开发者可以轻松地集成大模型功能，实现高效、智能的应用开发。

以盘古大模型为例，要调用盘古大模型的编程接口，需要开通相关服务并申请盘古大模型的使用权限，获取 API 地址统一资源定位器（Uniform Resource Locator, URL）和 API 密钥。然后使用 Python 的 requests 组件访问 API 地址，请求大模型提供服务。下面的代码是使用盘古大模型编程接

口获取服务的一个示例。

```
import requests import json
# 设置API 地址 URL 和请求头
url = "你的API 地址 URL"
headers = {
    "Content-Type": "application/json"
    "Authorization": "你的API 密钥"
}
# 定义请求数据
data = {
    "input": "请介绍盘古大模型编程接口的使用方法。",
    "max_length": 100
}
# 发送 POST 请求
response = requests.post(url, headers=headers, data=json.dumps(data))
#处理返回结果
if response.status_code == 200:
    result = response.json()
    print("大模型返回的结果:", result.get("output"))
else:
    print("请求失败:", response.status_code)
```

6.2 机器视觉

随着人工智能技术的创新和发展,许多知名公司开始致力于无人驾驶汽车的研究和设计。2014年,百度公司宣布启动自动驾驶研究项目,时至今日,百度 Apollo(阿波罗)已经经历了多次版本更新。2018年年底,Apollo 自动驾驶车队于长沙亮相,并在高速公路场景下演示了 L3/L4 级多车型车辆的协同,如图 6.10 所示。



图 6.10 百度 Apollo 自动驾驶车队

和人一样,无人驾驶汽车在学习经验的过程中也需要时刻观察周围的环境。虽然捕获图像并不难,但是对图像中的物体进行识别是一项具有挑战性的任务,这就涉及机器视觉技术。它利用计算机来模拟人的视觉,同时从图像中提取有用的信息并进行处理和理解。计算机实现人的视觉功能需要经过以下步骤。

1. 问题分解

要实现人的视觉功能,首先要获取图像。无人驾驶汽车通过安装在车身上的摄像头和传感器来获取图像。在设计无人驾驶汽车的时候,我们使用监督学习算法来使无人驾驶汽车学会识别和解析交通标志,并将识别结果与一组有标记的交通标志进行比较。计算机通过对摄像头捕获的图像进行特征分析,了解其中的一些重要的细节,从而在面对一个新的输入图像的时候实现精准地判断这一

图像中有哪些交通标志，并据此做出决策。

2. 模式识别

在无人驾驶汽车的示例中，车身上的摄像头就相当于人类的眼睛。通常人类在驾驶汽车的时候会观察道路上有没有障碍物，并且需要识别路标和信号灯；对于无人驾驶汽车来说，摄像头捕获的是图像，计算机需要利用神经网络来对图像中的重要信息进行分析，包括图像特征的定位、对比等，最终根据分析结果进行行进动作的决策。

无人驾驶汽车与人工驾驶汽车的不同之处如表 6.2 所示。

表 6.2 无人驾驶汽车与人工驾驶汽车的不同之处

无人驾驶汽车	人工驾驶汽车
用摄像头识别	用眼睛识别
用神经网络对捕获的图像进行分析从而做决策	根据观察到的事物用大脑做决策
用二值图像、灰度图像或 RGB 彩色图像的抽象方法对图像进行转化	观察物体的光学图像

3. 抽象

人类通过观察物体的光学图像来识别物体，那么计算机又是如何识别图像中的物体的呢？这里介绍 3 种计算机对图像进行抽象的方式：二值图像、灰度图像和 RGB 彩色图像。二值图像的二维矩阵仅由 0、1 两个值构成，“0”代表黑色，“1”代表白色，它通常用于文字、线条图的扫描识别；灰度图像矩阵元素的取值范围通常为[0, 255]，其中“0”表示纯黑色，“255”表示纯白色，中间的数字从小到大表示由黑到白的过渡色；RGB 彩色图像用红（R）、绿（G）、蓝（B）三原色的组合来表示每个像素的颜色，它利用 3 个 $M \times N$ 的二维矩阵分别表示各像素的 R、G、B 颜色分量，其中 M 、 N 分别表示图像中不同像素的行数和列数。

4. 算法设计

在把一个图像抽象成计算机可以识别的矩阵之后，就可以利用人工智能的算法对数据进行学习和处理了。在机器视觉领域，人工神经网络是最基本的算法。

(1) 人工神经网络与深度学习

机器视觉技术的广泛应用得益于人工神经网络（简称神经网络）的发展。神经网络是开发人工智能必不可少的工具之一，它受人的大脑工作机制的启发，揭示了人工智能的本质。人工神经网络模拟的是人脑的特征，因此它的发展也离不开现代神经科学的研究成果。一个神经网络是由大量的神经元依照某种方式互连而成的，每一个神经元都相当于一个简单的处理单元，它是神经网络的设计基础。

图 6.11 所示，生物神经元主要包括细胞体、树突、轴突和突触等结构。其中细胞体由细胞核、细胞质和细胞膜组成。神经元通过其输入端的树突接收从其他神经元传递过来的信号，并通过轴突传递这些信号。轴突末端的细小分支称为神经末梢。神经元通过突触相互连接。

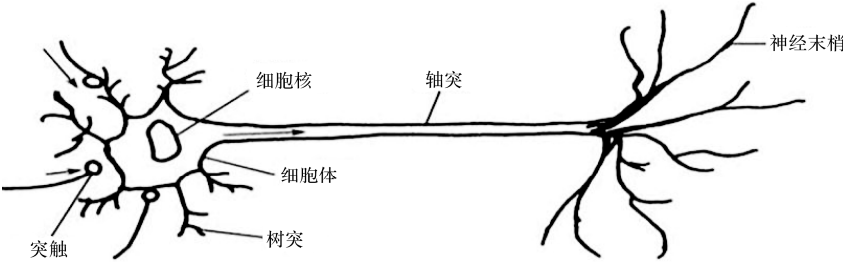


图 6.11 生物神经元结构图

人们根据生物神经元的结构提出了很多人工神经元模型，图 6.12 所示为一个麦卡洛克·皮茨

(McCulloch-PittsMP)模型。其中 $x_1, x_2, \dots, x_i, \dots, x_n$ 是来自前面 n 个神经元的信息； $w_1, w_2, \dots, w_i, \dots, w_n$ 是神经元连接权重，即神经元之间的突触连接强度； θ 是神经元的阈值。一个神经元根据与其他神经元之间的连接权重 $w_1, w_2, \dots, w_i, \dots, w_n$ 对输入的信息 $x_1, x_2, \dots, x_i, \dots, x_n$ 进行加权求和等处理，最终输出

$$y = f(\sum_{i=1}^n w_i x_i - \theta) \circ f(\cdot) \text{ 取阶跃函数 } \text{step}(\cdot), \text{ 有 } \text{step}(a) = \begin{cases} 1, & a \geq 0 \\ 0, & a < 0 \end{cases} \circ$$

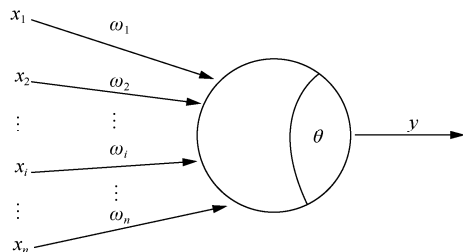


图 6.12 MP 模型

神经网络的一个典型应用是解决分类问题。在分类问题中，神经网络一般包含输入层、隐含层和输出层，其中输入层神经元数量由输入向量的维数决定，输出层神经元数量由需要分类的类别数量决定。输入层不对数据进行处理，仅将输入数据送入下一层进行运算。在输入层与输出层中间存在若干隐含层，这些隐含层主要用于对数据进行计算和处理。

图 6.13 所示为一个简单的前馈神经网络。神经网络由大量彼此连接的神经元组成，神经网络的类型由神经元的连接方式决定。一种常用的神经网络是多层前馈神经网络，其中每一层的神经元都与下一层完全互连，并且既没有同层连接，也没有跨层连接。输入层接受外部输入，隐含层和输出层的神经元处理信号，最后输出层的神经元输出结果。在学习过程中，神经网络根据训练数据调整神经元之间的连接权重和每个功能神经元的阈值。

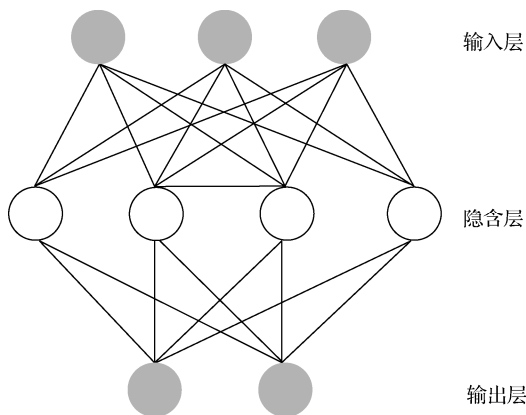


图 6.13 一个简单的前馈神经网络

神经网络的工作过程包括学习期和工作期。在学习期，每个处理单元通过学习样本（根据特定算法）来修改连接权重。在工作期，每个连接权重是固定的，处理单元的状态会发生变化，最后会达到稳定状态。

值得一提的是，即便一个神经网络只有一个隐含层，其输出也会以任意精度逼近任意连续函数，如果增加网络的深度及神经元数量，就能够建立更为复杂的模型，也就是深度学习模型。区别于传统浅层学习，深度学习更强调模型结构的深度，通常隐含层节点有 5 层、6 层，甚至更多。但层次的增加会带来很多问题，最主要的问题就是梯度消失，这也是众多研究者致力于解决的问题。

目前,比较典型的深度学习模型有卷积神经网络、深度信念网络和堆栈自编码网络等,接下来介绍深度学习模型之一——卷积神经网络。

(2) 卷积神经网络

卷积神经网络是在1989年被提出的一种深度神经网络。一个传播较为广泛的卷积神经网络是纽约大学扬·勒丘恩(Yann LeCun)教授在1998年提出的LeNet-5,其结构如图6.14所示,主要包括输入层(Input层)、卷积层(C1层、C3层和C5层)、池化层(S2层和S4层)、全连接层(F6层)和输出层(Output层)。如今,卷积神经网络已被广泛应用于机器视觉等领域,并且在很多问题上具备出色的性能。

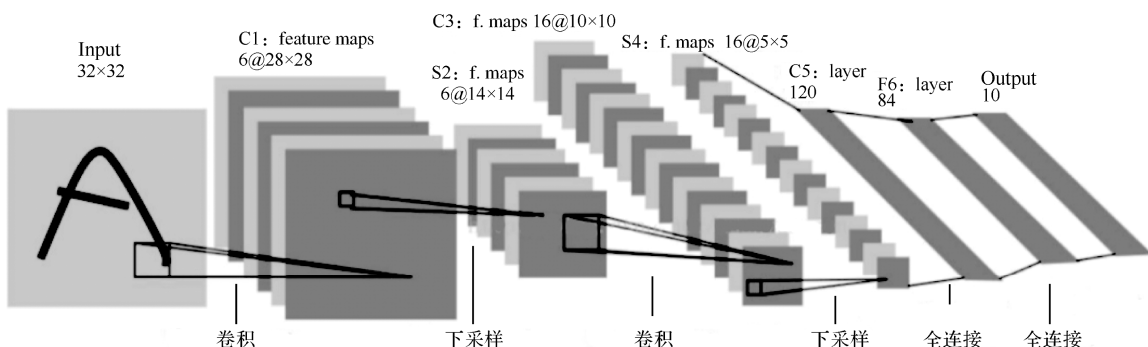


图 6.14 典型的卷积神经网络 LeNet-5 的结构

卷积神经网络利用卷积层和池化层来学习图像在各个尺度上的特征,这是受人类认知图像过程启发的结果。人类认知图像的过程是有一定层次的:首先,理解图像的颜色与亮度;其次,分辨边角线等局部细节特征;再次,关注更复杂的信息和结构,如形状、纹理等;最终综合以上观察结果形成整个物体的概念。

如图6.15所示,卷积神经网络由输入层、卷积层、池化层、全连接层和输出层构成。

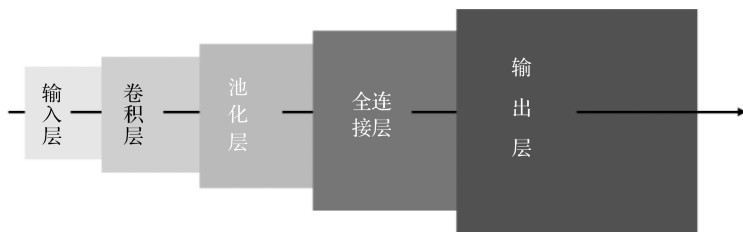


图 6.15 一个简单的卷积神经网络

输入层可以处理多维数据。卷积神经网络广泛应用于机器视觉领域,在该领域中网络的输入数据一般是平面上的像素点所对应的RGB通道,因此一般是三维的。卷积神经网络的学习使用的是梯度下降法,因此需要对输入数据进行标准化处理。具体而言,对于分布在[0,255]的像素值,需要进行归一化处理,以提升网络的学习效率。

卷积层主要对输入数据进行特征提取。一个卷积层包含多个卷积核,每个卷积核都是一个特征提取器。与前馈神经网络中的神经元类似,组成卷积核的每个元素都对应一个权重和一个偏差量。一个卷积核对图像是有一定的卷积范围的,这一范围称为“感受野”,对应于视觉皮层细胞的感受野。如果要在不同的尺度上进行特征提取,但需要增加卷积层的数量。

虽然卷积层可以对输入的图像进行特征抽取和降维,但是特征图像的维数还是会很高,这容易导致过拟合,且计算起来非常耗时。