

人工智能在计算机应用软件开发中的应用探索

郑建波

绵阳市绵州通科技有限责任公司，四川 绵阳 621000

DOI: 10.61369/TACS.2026040008

摘 要： 当下计算机应用软件开发面临效率瓶颈，代码复用性差、需求变更响应迟缓，且传统开发模式难以精准适配复杂多变的业务场景，导致软件质量与交付周期难以保障。基于此，本文深入探究人工智能在计算机应用软件开发中的创新应用路径。通过引入智能代码生成、自动化测试、需求智能解析等技术，旨在打破传统开发局限，提升开发效率与代码质量，实现软件快速迭代与精准交付，为计算机应用软件开发领域提供智能化转型新思路。

关 键 词： 人工智能；计算机应用软件；自然语言处理

Exploration of the Application of Artificial Intelligence in Computer Application Software Development

Zheng Jianbo

Mianyang Mianzhoutong Technology Co., Ltd., Mianyang, Sichuan 621000

Abstract： Currently, computer application software development is facing efficiency bottlenecks, such as poor code reusability, slow response to requirement changes, and the difficulty of traditional development models in accurately adapting to complex and volatile business scenarios, which makes it hard to guarantee software quality and delivery cycles. Based on this, this paper deeply explores the innovative application paths of artificial intelligence (AI) in computer application software development. By introducing technologies such as intelligent code generation, automated testing, and intelligent requirement analysis, it aims to break the limitations of traditional development, improve development efficiency and code quality, realize rapid software iteration and accurate delivery, and provide new ideas for the intelligent transformation in the field of computer application software development.

Keywords： artificial intelligence (AI); computer application software; natural language processing

引言

“人工智能 + 制造”专项行动实施意见明确指出以习近平新时代中国特色社会主义思想为指导，深入贯彻党的二十大和二十届历次全会精神，完整准确全面贯彻新发展理念，加快构建新发展格局，统筹发展和安全，坚持创新驱动、场景牵引、市场主导、安全可信、开放共享、普惠融通，一端抓技术供给，推动“智能产业化”，一端抓赋能应用，加快“产业智能化”，整体壮大产业生态，促进人工智能科技创新与产业创新深度融合、人工智能技术与制造业应用“双向赋能”，加快制造业智能化、绿色化、融合化发展，有力支撑制造强国、网络强国和数字中国建设^[1]。企业应该根据国家的政策性文件走符合国家发展的道路，这样才能够促进自身的发展。

一、人工智能在计算机应用软件开发中应用的意义

（一）提升开发效率与质量

在计算机应用软件开发中，人工智能的应用极大地提升了开发效率与质量。从需求分析环节来看，自然语言处理技术能精准解析用户以自然语言表述的需求，快速提取关键信息，减少人工理解偏差，缩短需求确认周期。设计阶段，智能架构设计算法依据软件特性与需求，迅速生成适配的架构方案，避免人工设计可

能出现的架构不合理问题^[2]。编码时，代码自动生成与智能补全功能，减少开发人员重复编码工作，降低出错率。测试环节，智能测试用例生成覆盖更多边界情况，缺陷智能定位与分类能快速精准找出问题，保障软件质量，整体推动开发进程高效优质推进^[3]。

（二）增强软件智能化与个性化

人工智能赋予计算机应用软件更强的智能化与个性化能力。借助机器学习算法，软件能够自主学习用户行为模式与偏好，依据这些信息为用户提供个性化服务。例如，根据用户历史操作习

惯优化界面布局与功能展示顺序，提升用户操作的便捷性与舒适度^[4]。在交互方面，自然语言处理和语音识别技术使软件能以更自然的方式与用户交流，理解用户意图并准确回应。同时，智能决策系统让软件在面对复杂情况时能自主分析并做出合理决策，为用户提供更智能、贴心的服务体验，满足用户日益多样化、个性化的需求，增强软件的市场竞争力^[5]。

二、人工智能在计算机应用软件开发的应用

（一）需求分析阶段

1. 自然语言处理助力需求理解

在传统的需求分析过程中，开发人员是通过与用户进行面对面沟通或查阅文档的方式来了解用户的需求，这导致信息传递可能存在一些理解性的偏差。现如今，开发人员可通过人工智能当中的自然语言处理技术来对用户的需求进行语义分析和理解，从而更好地了解用户的真实想法^[6]。例如，开发人员可利用词法分析、句法分析等技术，来将用户的需求语句分解为单词和短语，并分析它们之间的语法关系，从而更好地了解用户需求的含义；还可通过语义角色标注等方法，进一步确定用户需求的语义内容；更可通过情感分析技术，来分析用户对需求的重视程度^[7]。

2. 智能需求挖掘与预测

开发人员可对用户大量的历史需求数据和相关项目资料进行分析和挖掘，从而更好地发现潜在的需求模式和趋势。例如，开发人员可通过机器学习中的分类算法，将用户已有的需求按照不同的特征进行分类，从而进一步了解各类需求的特点和规律；还可通过预测算法来预测当前项目的相关信息和需求，从而更好地指导后续的开发工作；更可以根据这些分析结果，提前细化用户的需求，从而满足用户的潜在需求。

（二）设计阶段

1. 智能架构设计

软件架构设计是软件工程中非常重要的步骤，软件架构决定软件的性能、扩展性和可维护性。人工智能可以依据软件的功能需求和性能指标、流量数据等因素，分析已有的架构成功案例，用智能推荐的方式从已知的架构案例中推荐出适合的架构方案给软件开发人员。例如，对于一个大型的分布式系统，人工智能可以分析该系统的业务逻辑、数据分布和访问方式，推荐采用微服务的软件架构^[8]。采用微服务架构，可以将一个大型的分布式系统划分为多个小型的独立的服务，每个服务都是独立的，可以独立开发、部署、扩展，提高了系统的灵活性和可维护性。人工智能还可以根据系统性能指标要求，分析推荐采用哪种分布式的数据库来搭建，以及哪些缓存策略来优化系统性能^[9]。

2. 个性化界面设计

开发人员应该在了解用户需求的基础上，进行个性化的界面设计，这样才能够给客户留下更好的印象，与客户进行后续的深入沟通和交流。例如，开发人员可根据用户的行为数据、偏好和审美观念，实现个性化的界面设计；也可以收集用户使用类似软件时的操作习惯、点击热区、停留时间等数据，来分析用户的行

为和偏好，从而更好地为用户进行服务；更可以根据用户的反馈意见，持续优化界面设计，提升用户的使用体验。

（三）编码阶段

1. 代码自动生成

开发人员在传统编写代码的过程中会通过查找资源、借鉴相似网站的方式来进行代码的编写。现如今，开发人员可通过自然语言处理技术将设计文档中的功能描述转化为计算机可理解的代码逻辑，再结合代码模板和智能填充算法生成具体的代码，并进行一定的优化和调整，以此来找到适合本程序的代码，减少人工编写代码的工作量和错误率。例如，开发人员要实现一个包括用户名和密码的输入、验证以及登录成功后的跳转用户登录功能，可以将文字输入到人工智能系统当中，并让其转化为代码的框架，填充一些常见的输入验证的代码、数据库查询的代码等代码片段，进行适当的修改和完善，最终查看展示的界面效果，以此来节省更多的代码编写时间。

2. 智能代码补全与纠错

开发人员在传统编写代码并查找错误的过程中，通过设置断点的方式来进一步定位错误，这会消耗大量的时间。现如今，开发人员可将代码输入人工智能当中，并让其分析代码的上下文和语法结构，预测开发人员接下来可能要输入的代码，并提供相应的补全建议；还可利用静态代码分析技术，来检测代码中可能存在的语法错误、逻辑错误和潜在的安全漏洞，从而及时解决出现的问题。例如，开发人员输入一个函数调用的部分代码，此时人工智能会自动补全函数的参数列表，并分析代码中可能存在的空指针引用、数组越界等常见错误，指出错误的位置及原因，提供修改的建议，从而使他们能够快速地解决问题。

（四）测试阶段

1. 智能测试用例生成

开发人员在传统的测试当中主要是通过人工设计用例，这不仅会出现效率较低的问题，还会出现覆盖面不全的问题。现如今，开发人员可通过遗传算法、模糊测试等技术来生成符合软件功能需求和代码结构的测试用例，从而更好地提高测试的全面性和有效性，快速地发现可能存在的问题。例如，对于一个输入参数为整数的函数，开发人员可通过人工智能来生成包含正整数、负整数、零、极大值和极小值等各种边界值的测试用例，以及一些随机生成的异常值测试用例，从而自动地检测函数在不同输入情况下的稳定性和正确性，提高测试的效率和质量。

2. 缺陷智能定位与分类

开发人员可通过人工智能技术来分析测试结果数据和代码执行轨迹，并利用机器学习算法建立缺陷预测模型，将缺陷与代码中的特定模块、函数或语句关联起来，从而快速准确地定位软件缺陷的位置和类型。例如，人工智能发现一个测试用例执行失败，既会分析该测试用例的输入数据、预期输出和实际输出，又会分析代码的执行路径和变量值等信息，从而更好地确定缺陷可能存在的代码段。同时，人工智能会根据功能缺陷、性能缺陷、安全缺陷等缺陷进行分类，并提供针对性的修复建议。

（五）维护阶段

1. 智能故障诊断与预测

开发人员可通过人工智能来实时监测软件的性能指标和运行状态，以此来更好地建立故障预测模型，发现潜在的故障风险，进行故障诊断，确定故障的原因和位置。例如，开发人员对于一个服务器端的软件系统，可以通过人工智能监测服务器的 CPU 使用率、内存占用率、网络流量等指标，以此来更好地了解指标出现的异常波动情况，及时发出预警信号。其中，如果系统出现崩溃或响应缓慢等故障，人工智能会通过分析系统日志、错误信息等数据，结合故障预测模型，来快速定位有故障的问题，从而更好地维护系统的安全。

2. 软件智能升级与优化

随着业务的发展和用户需求的变化，软件需要不断进行升级和优化。人工智能可以根据用户反馈和使用数据，分析软件的性能瓶颈和功能不足之处，为软件升级提供决策支持。同时，在软件升级过程中，人工智能可以自动检测代码的兼容性问题，确保

升级后的软件能够稳定运行。例如，通过对用户对软件不同功能的使用频率和满意度的调查分析，人工智能可以确定哪些功能需要优先升级和优化。在升级过程中，利用代码分析技术检查新代码与旧代码之间的接口兼容性、数据类型兼容性问题，提前发现并解决潜在的兼容性问题，减少升级过程中的风险和故障^[10]。

三、结束语

在计算机应用软件开发不断迈向新阶段的当下，人工智能的融入已成为不可阻挡的趋势。它不仅为开发过程带来了效率的飞跃，更在代码优化、需求精准捕捉、智能测试等方面展现出巨大潜力，有力推动软件向智能化、个性化发展。然而，这一融合之路并非坦途，数据安全、算法可解释性等问题亟待解决。随着技术持续突破与完善，人工智能与计算机应用软件开发结合将愈发紧密。未来，开发人员需紧跟时代步伐，积极探索创新应用模式，让人工智能更好地赋能软件开发，为社会创造更多价值，开启计算机应用软件发展的崭新篇章。

参考文献

- [1] 赵霏. 基于神经网络技术的计算机应用软件开发技术探讨 [J]. 互联网周刊, 2024, (23): 28–30.
- [2] 张军. 基于人工智能的计算机应用软件开发技术分析 [J]. 软件, 2024, 45(10): 175–177.
- [3] 王晓红. 基于人工智能的计算机应用软件开发技术 [J]. 中国信息界, 2024, (05): 37–39.
- [4] 杨义. 基于动态库的嵌入式训练计算机应用软件移植的研究 [C]// 天津市电子工业协会. 天津市电子工业协会 2024 年年会论文集. 天津: 天津七一二通信广播股份有限公司; , 2024: 31–35.
- [5] 梁一笑. 浅析人工智能技术在计算机应用软件开发中的应用 [J]. 信息与电脑 (理论版), 2024, 36(08): 59–61.
- [6] 尹杰. 计算机应用软件自动化开发技术分析 [J]. 信息记录材料, 2024, 25(03): 165–167.
- [7] 肖转红. 人工智能在计算机应用软件设计中的应用 [J]. 电子技术, 2024, 53(02): 134–135.
- [8] 吕丰秀. 计算机应用软件中的自动化开发技术分析 [J]. 集成电路应用, 2023, 40(10): 150–151.
- [9] 李大伟. 基于人工智能的计算机应用软件开发技术分析 [J]. 软件, 2023, 44(09): 124–126.
- [10] 薛梦丹. 基于人工智能的计算机应用软件开发技术应用分析 [J]. 中国高科技, 2023, (13): 40–42.