

УДК 004.8

## ИСПОЛЬЗОВАНИЕ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА ДЛЯ УЛУЧШЕНИЯ КАЧЕСТВА РАЗРАБОТКИ И СОПРОВОЖДЕНИЯ ЦИФРОВЫХ СИСТЕМ

**Манукян Арсен Вардкесович**, магистрант, направление подготовки 09.04.01 Информатика и вычислительная техника, Казанский государственный энергетический университет, Казань, Россия  
e-mail: janegov@yandex.ru

Научный руководитель: **Сибеева Гульназ Рашитовна**, кандидат экономических наук, доцент кафедры цифровых систем и моделей, Казанский государственный энергетический университет, Казань, Россия  
e-mail: sibaeva.gr@kgeu.ru

***Аннотация.** Современные цифровые системы характеризуются высокой степенью сложности, что повышает требования к качеству, надёжности и скорости разработки программного обеспечения [2]. Традиционные методы контроля качества и сопровождения оказываются недостаточно эффективными в условиях роста объёмов данных и усложнения архитектур. Целью данной работы является анализ применения искусственного интеллекта для повышения качества разработки и сопровождения цифровых систем. В исследовании использованы методы анализа научных публикаций, сравнительного анализа и классификации подходов к использованию искусственного интеллекта (ИИ) в инженерии программного обеспечения. Показано, что применение ИИ повышает точность анализа кода, расширяет возможности тестирования, обеспечивает предиктивный мониторинг и автоматизацию технической поддержки. Полученные результаты подтверждают практическую значимость ИИ для повышения устойчивости и эффективности цифровых систем.*

***Ключевые слова:** искусственный интеллект, цифровые системы, сопровождение, качество программного обеспечения, машинное обучение.*

***Для цитирования:** Манукян А. В. Использование искусственного интеллекта для улучшения качества разработки и сопровождения цифровых систем // Шаг в науку. – 2026. – № 1. – С. 8–11.*

## USING ARTIFICIAL INTELLIGENCE TO IMPROVE THE QUALITY OF DEVELOPMENT AND MAINTENANCE OF DIGITAL SYSTEMS

**Manukyan Arsen Vardkesovich**, postgraduate student, training program 09.04.01 Computer Science and Engineering, Kazan State Power Engineering University, Kazan, Russia  
e-mail: janegov@yandex.ru

Research advisor: **Sibaeva Gulnaz Rashitovna**, Candidate of Economic Sciences, Associate Professor of the Department of Digital Systems and Models, Kazan State Power Engineering University, Kazan, Russia  
e-mail: sibaeva.gr@kgeu.ru

***Abstract.** Modern digital systems are characterized by a high level of complexity, which increases the requirements for the quality, reliability, and speed of software development. Traditional methods of quality control and maintenance prove to be insufficiently effective under conditions of growing data volumes and increasing architectural complexity. The purpose of this paper is to analyze the application of artificial intelligence to improve the quality of development and maintenance of digital systems. The study employs methods of scientific literature analysis, comparative analysis, and classification of approaches to the use of artificial intelligence in software engineering. It is shown that the application of artificial intelligence increases the accuracy of source code analysis, expands testing capabilities, provides predictive monitoring, and enables the automation of technical support. The obtained results confirm the practical significance of artificial intelligence in enhancing the stability and efficiency of digital systems.*

***Key words:** artificial intelligence, digital systems, maintenance, software quality, machine learning.*

***Cite as:** Manukyan, A. V. (2026) [Using artificial intelligence to improve the quality of development and maintenance of digital systems]. *Shag v nauku* [Step into science]. Vol. 1, pp. 8–11.*



## Введение

Развитие цифровых технологий привело к появлению многоуровневых и распределённых программных систем, требующих непрерывного контроля качества, адаптации и совершенствования. Рост объёмов данных, увеличение количества микросервисов, сложность интеграций и высокая скорость изменений создают значительные сложности для специалистов, занимающихся разработкой и сопровождением программного обеспечения [4]. В условиях возрастающей нагрузки традиционные методы тестирования, мониторинга и анализа состояния системы становятся недостаточно эффективными.

Искусственный интеллект способен автоматизировать рутинные операции, выявлять паттерны в больших массивах данных, прогнозировать потенциальные сбои и помогать разработчикам в принятии решений. ИИ применяется для анализа кода, автоматизации тестирования, предиктивного мониторинга, обработки логов, классификации инцидентов и взаимодействия с пользователями. Такое использование повышает устойчивость цифровых систем и уменьшает объём человеческих ошибок [5].

Целью данной статьи является анализ применения искусственного интеллекта для повышения качества разработки и сопровождения цифровых систем, определение ключевых направлений использования ИИ.

В процессе исследования применялись следующие методы.

Анализ научной литературы – изучение современных работ, посвящённых применению ИИ в инженерии ПО, DevOps и AIOps [7].

Исследование инструментов – изучение функциональности современных платформ анализа программного кода (например, Codiga AI, DeepCode), систем автоматизированной генерации тестов, а также AIOps-платформ, применяемых для мониторинга и сопровождения цифровых систем. Анализ включал рассмотрение используемых алгоритмов машинного обучения, областей применения инструментов, степени автоматизации процессов и их влияния на качество программного обеспечения.

Таблица 1. Основные направления применения ИИ

Направление	Примеры	Эффект
Анализ кода	Поиск ошибок, рекомендаций	Повышение качества
Тестирование	Генерация тестов	Экономия времени
AIOps	Анализ логов, прогноз сбоев	Уменьшение аварий
Поддержка	Автоответы, чат-боты	Снижение нагрузки

Источник: составлено автором

Сравнительный анализ – сопоставление традиционных и интеллектуальных методов тестирования и мониторинга с точки зрения эффективности выявления дефектов, скорости обработки данных, уровня автоматизации и зависимости от человеческого фактора. Особое внимание уделялось различиям в подходах к обнаружению ошибок, прогнозированию сбоев и реагированию на инциденты в процессе эксплуатации цифровых систем.

Метод классификации – систематизация существующих подходов к применению искусственного интеллекта в разработке и сопровождении программного обеспечения и выделение основных направлений, в которых ИИ оказывает наиболее значимое влияние на качество цифровых систем. Классификация проводилась на основе анализа функциональных возможностей ИИ-инструментов и их роли на различных этапах жизненного цикла программного продукта.

Системный подход – оценка взаимосвязи между внедрением интеллектуальных технологий и общим качеством разработки и сопровождения цифровых систем, включая влияние ИИ на надёжность, сопровождаемость, устойчивость и предсказуемость функционирования программного обеспечения. Рассматривались как технические аспекты внедрения ИИ, так и организационные факторы, связанные с изменением процессов разработки и эксплуатации.

Применение указанных методов позволило сформировать объективное и комплексное представление о роли искусственного интеллекта в современных цифровых системах, а также выявить ключевые области, в которых использование ИИ обеспечивает наибольший практический эффект.

В результате анализа научной литературы и практических решений в области применения искусственного интеллекта в разработке и сопровождении цифровых систем были выделены основные направления его использования, оказывающие наибольшее влияние на качество программного обеспечения. Обобщённая характеристика выделенных направлений представлена в таблице 1.

*Интеллектуальный анализ программного кода.* Установлено, что ИИ-системы позволяют выявлять ошибки, уязвимости и архитектурные анти-паттерны на основе анализа больших массивов исходного кода. В отличие от традиционных статических анализаторов, такие системы способны обнаруживать скрытые закономерности и формировать рекомендации по улучшению структуры программного обеспечения, что способствует повышению его качества [6].

*Генерация и оптимизация тестирования.* Применение ИИ в тестировании обеспечивает автоматическую генерацию тестовых сценариев и позволяет определить участки кода с повышенным риском воз-

никновения дефектов. Это повышает эффективность регрессионного тестирования и существенно сокращает затраты времени и ресурсов на обеспечение качества программного продукта.

*Предиктивный мониторинг (AIOps).* Результаты показали, что использование ИИ для анализа логов, метрик и событий позволяет выявлять аномалии в работе системы и прогнозировать возможные сбои до их возникновения. Такой подход снижает вероятность аварий и обеспечивает более стабильную эксплуатацию цифровых сервисов по сравнению с традиционным мониторингом, что наглядно отражено в таблице 2 и подтверждается результатами современных исследований.

Таблица 2. Сравнение традиционных и ИИ-подходов

Процесс	Традиционный подход	Подход с ИИ
Анализ кода	Ручное ревью, статический анализ	Модели ML находят скрытые ошибки
Тестирование	Ручное и скриптовое	Автоматическая генерация тестов
Мониторинг	Реакция после сбоя	Прогноз сбоев заранее
Поддержка	Операторы вручную	Чат-боты и авто-классификация

*Источник: составлено автором*

*Автоматизация технической поддержки.* Установлено, что ИИ-инструменты эффективно обрабатывают обращения пользователей, выполняя классификацию запросов и предлагая типовые решения. Это снижает нагрузку на специалистов технической поддержки и сокращает время реакции на инциденты [3].

В целом результаты подтверждают, что применение искусственного интеллекта позволяет повысить качество разработки, надёжность и сопровождаемость цифровых систем, а также повысить эффективность ключевых процессов их жизненного цикла.

Результаты исследования подтверждают, что ИИ значительно улучшает процессы разработки и сопровождения цифровых систем. Интеллектуальные инструменты обеспечивают более глубокий анализ данных, снижают зависимость от человеческого фактора и повышают уровень автоматизации. Однако существует ряд ограничений: качество прогнозов зависит от полноты обучающих данных, интеграция ИИ требует инфраструктурных изменений, а внедрение требует подготовки специалистов.

Исследования других авторов показывают аналогичные тенденции. Например, современные работы в области AIOps подтверждают эффективность предиктивного анализа логов, а аналитика в тестировании демонстрирует устойчивое улучшение покрытия при генерации сценариев на основе ИИ [1].

Несмотря на возможные отклонения результатов и редкие ложные срабатывания, преимущества ИИ значительно превышают недостатки. Технологии искусственного интеллекта постепенно становятся стандартом для цифровой инженерии.

Проведённый анализ показал, что применение искусственного интеллекта в разработке и сопровождении цифровых систем оказывает комплексное и устойчивое положительное влияние на их качество и надёжность. Использование ИИ позволяет автоматизировать рутинные операции, улучшать анализ исходного кода, расширять возможности тестирования и повышать точность мониторинга. Интеллектуальные инструменты сокращают вероятность ошибок, снижают технический долг и улучшают предсказуемость работы программных продуктов.

Сравнение традиционных подходов с моделями на основе машинного обучения подтверждает, что ИИ обеспечивает более высокую скорость обработки данных и способность выявлять скрытые закономерности, недоступные человеку.

Вместе с тем внедрение ИИ связано с необходимостью качественного сбора данных, корректной настройки моделей и адаптации рабочих процессов. Эти факторы остаются ключевыми ограничениями технологии. Однако, несмотря на существующие трудности, тенденции развития цифровых систем указывают

на дальнейшее расширение применения ИИ и усиление его роли в инженерии программного обеспечения.

Таким образом, искусственный интеллект является перспективным и практически значимым ин-

струментом, способствующим повышению качества разработки и сопровождения цифровых систем и формирующим основу для построения более устойчивых, предсказуемых и эффективных цифровых решений.

### Литература

1. Использование искусственного интеллекта для разработки программного обеспечения / Д. А. Тикки [и др.] // Международный журнал информационных технологий и энергоэффективности. – 2023. – № 12(38). – С. 64–71. – EDN: DZMXKT.
2. Смирнов Ю. Н. Основы проектирования и разработки цифровых платформ предприятий // Вестник Казанского государственного технического университета им. А.Н. Туполева. – 2018. – Т. 74, № 3. – С. 155–161. – EDN: YTHZOX.
3. Соколова Э. С., Жолобов Д. М. Применение искусственного интеллекта для генерации тестовых данных при автоматизации тестирования программного обеспечения // Математические методы в технологиях и технике. – 2023. – № 6. – С. 88–91. – [https://doi.org/10.52348/2712-8873\\_MMTT\\_2023\\_6\\_88](https://doi.org/10.52348/2712-8873_MMTT_2023_6_88). – EDN: LJMOKA.
4. Фомин А. В. Роль предметно-ориентированного проектирования в микросервисной архитектуре на примере автоматизированной системы краудсорсинга // Тинчуринские чтения – 2024 «Энергетика и цифровая трансформация»: Материалы Международной молодежной научной конференции, Казань, 24–26 апреля 2024 года. – Казань: Казанский государственный энергетический университет, 2024. – С. 110–113. – EDN: NSOZCS.
5. Adhyapak S. (2024) Data Privacy and Security Risks in AI-Based Code Understanding. *International Journal for Research in Applied Science and Engineering Technology*. – Vol. 12. – No. 6, pp. 1913–1921. – <https://doi.org/10.22214/ijraset.2024.63423>.
6. Hellendoorn V. J., Sawant A. A. (2022) The growing cost of deep learning for source code. *Association for Computing Machinery. Communications of the ACM*. – Vol. 65. – No. 1, pp. 31–33. – <https://doi.org/10.1145/3501261>.
7. Madabushini P. M. (2025) The evolution of self-healing infrastructure: AI-driven automation in modern devops. *International Journal of Information Technology and Management Information Systems*. – Vol. 16. – No. 2, pp. 891–899. – [https://doi.org/10.34218/ijitmis\\_16\\_02\\_058](https://doi.org/10.34218/ijitmis_16_02_058).

Статья поступила в редакцию: 18.12.2025; принята в печать: 27.02.2026.

Автор прочитал и одобрил окончательный вариант рукописи.