УДК 37.06

ИНТЕГРАЦИЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА В ДИСЦИПЛИНУ «ИНФОРМАТИКА»

Чугузов Е.П.

старший преподаватель

Алтайский государственный аграрный университет,

Барнаул, Россия

Аннотация. В статье рассматривается системное внедрение технологий искусственного интеллекта (ИИ) в образовательный процесс дисциплины «Информатика». Анализируется переход от традиционной алгоритмической парадигмы обучения к концепции «человекоцентричного ИИ», где нейросети выступают не объектом изучения, а когнитивным инструментом. Обосновывается необходимость формирования у обучающихся компетенций XXI века: критического мышления при работе с генеративными моделями, навыков промпт-инжиниринга и понимания этических ограничений алгоритмов. Подробно разбираются практические кейсы применения ИИ на разных уровнях образования: от визуального программирования и создания чат-ботов в средней школе до использования LLM (Large Language Models) для отладки кода, генерации SQL-запросов и анализа данных в старшей школе и вузе. Сделан вывод о том, что ИИ становится «вторым пилотом» для преподавателя, позволяя автоматизировать рутину и сосредоточиться на развитии творческого потенциала студентов.

Ключевые слова: искусственный интеллект, информатика, машинное обучение, LLM, генеративные модели, промпт-инжиниринг, адаптивное обучение, цифровая грамотность, педагогика.

INTEGRATION OF ARTIFICIAL INTELLIGENCE INTO THE DISCIPLINE "INFORMATICS"

Chuguzov E.P.
Senior Lecturer
Altai State Agrarian University,
Barnaul, Russia

Abstract. The article examines the systematic implementation of artificial intelligence (AI) technologies in the educational process of the "Informatics" discipline. It analyzes the shift from the traditional algorithmic paradigm of teaching to the concept of "human-centered AI," where neural networks are not an object of study but a cognitive tool. The article substantiates the need to develop 21st-century competencies in students: critical thinking when working with generative models, skills in prompt engineering, and an understanding of the ethical limitations of algorithms. Practical cases of AI application at different levels of education are examined in detail: from visual programming and creating chatbots in secondary school to using LLM (Large Language Models) for code debugging, generating SQL queries, and data analysis in high school and university. It is concluded that AI is becoming a "second pilot" for the teacher, allowing them to automate routine tasks and focus on developing students' creative potential.

Keywords: искусственный интеллект, информатика, машинное обучение, LLM, генеративные модели, промпт-инжиниринг, адаптивное обучение, цифровая грамотность, педагогика.

Дисциплина «Информатика» исторически являлась зеркалом технологического прогресса. В 80-е годы она учила печатать и работать с DOS, в 90-е — осваивать Windows и Office, в 2000-е — писать код на Pascal и C++. Сегодня мы находимся в точке бифуркации: появление доступных больших языковых моделей (LLM) и мультимодальных нейросетей требует

фундаментального пересмотра учебных планов. Информатика больше не может быть просто наукой о «железе» и алгоритмах. Она должна стать наукой о взаимодействии человека и искусственного интеллекта.

Актуальность темы обусловлена тем, что ИИ перестал быть абстрактной технологией будущего — он стал частью повседневной жизни и рабочего инструментария. Для преподавателя информатики сегодня критически важно не просто познакомить учащихся с синтаксисом Python, но и сформировать у них компетенции работы с ИИ-инструментами, чтобы они не стали жертвами автоматизации, а стали её архитекторами [1].

Традиционное обучение программированию строится на модели «пустого листа»: студент должен написать код с нуля, помня синтаксис и библиотеки. ИИ-инструменты (такие как GitHub Copilot, GigaChat, ЯндексGPT) меняют этот подход. Теперь базовой компетенцией становится не скорость набора символов, а умение правильно поставить задачу машине и проверить результат.

В контексте дисциплины «Информатика» это означает переход к трем новым навыкам:

1. **Промпт-инжиниринг (Prompt Engineering).** Умение формулировать запросы к нейросети так, чтобы получить оптимальный код. Это требует четкого понимания логики, которую студент хочет реализовать.
2. **Верификация и отладка (Debugging).** ИИ часто пишет код, который работает, но неэффективен или содержит скрытые уязвимости. Задача студента — проанализировать предложенный вариант, найти слабые места и оптимизировать его.
3. **Этическая рефлексия.** Понимание того, какие данные использует модель, нет ли в коде предвзятости (bias) и не нарушает ли решение законы о персональных данных [1].

2. Практические подходы к внедрению ИИ по уровням образования

Интеграция ИИ должна быть поэтапной, соответствующей когнитивным возможностям обучающихся.

2.1. Начальный этап (5–7 классы): Визуальное программирование и концептуальное понимание На этом этапе важно не перегружать детей синтаксисом. Цель — показать, как «обучается» машина.

- **Инструмент** Scratch с расширениями для ИИ, Code.org AI Labs.
- **Кейс «Распознавание жестов»:** Учащиеся с помощью веб-камеры обучают модель различать три жеста (кулак, ладонь, «окей»). Они видят, как нейросеть строит датасет, и понимают, что для точности нужно 50–100 фотографий каждого жеста. Это наглядно объясняет понятие «обучающая выборка».
- **Кейс «Чат-бот в Scratch»:** Создание простого бота, который реагирует на ключевые слова. Это знакомит с логикой «Если... То...» в контексте ИИ.

2.2. Средний этап (8–9 классы): Основы алгоритмики и анализа данных на Python Здесь вводится текстовое программирование, но ИИ выступает как помощник в написании шаблонного кода.

- **Инструменты:** Python (библиотеки Scikit-learn, Pandas), Jupyter Notebook.
- **Кейс «Линейная регрессия»:** Задание на прогнозирование оценки за контрольную. Учащийся собирает данные (время подготовки, посещаемость), а ИИ помогает написать каркас кода:
- **Кейс «Фильтрация спама»:** Использование алгоритма «Наивного Байеса». Студент учится предобрабатывать текст (удалять стоп-слова), а ИИ помогает с имплементацией библиотеки.

2.3. Продвинутый этап (10–11 классы и вузы): Генеративный ИИ и сложные архитектуры Учащиеся сами становятся «кураторами» нейросетей.

- **Инструменты:** API Llama 3, GigaChat, GitHub Copilot, Kaggle.
- **Кейс «Генерация SQL-запросов»:** Учащийся описывает словами, какие данные ему нужны из базы («Выбрать имена клиентов, которые купили

товар А в марте»), а ИИ генерирует сложный SQL-запрос с JOIN. Студент должен проверить его корректность.

- **Кейс «Компьютерное зрение для экологии»:** Проект по подсчету птиц на фото. Студенты размечают датасет, обучают модель YOLO v8 и анализируют точность (Precision/Recall) [2].

3. Роль ИИ как ассистента преподавателя (Teacher's Copilot)

ИИ кардинально снижает нагрузку на преподавателя информатики, который часто работает с большими группами:

- **Генерация индивидуальных заданий.** Нейросеть может создать 30 уникальных вариантов контрольной работы по теме «Циклы в Python», каждый со своим условием, но одинаковым уровнем сложности. Это исключает списывание.
- **Мгновенная проверка кода.** ИИ-ассистент может анализировать код, написанный учащимися, указывать на синтаксические ошибки, предлагать более «питонические» способы решения (например, заменить цикл на list comprehension) и объяснять, почему возникла ошибка IndexError.
- **Создание интерактивных симуляторов.** ИИ позволяет генерировать виртуальные среды для тестирования кода, где учащийся может видеть, как меняются переменные в памяти (визуализация стека и кучи) [2].

4. Этические дилеммы и риски интеграции

Внедрение ИИ в информатику сопряжено с серьезными вызовами, которые сами по себе являются темами для занятий:

- **Риск «отупения».** Если учащийся просто копирует код от ИИ, он не учится. Необходимо внедрять правило: «Сначала напиши план решения сам, потом попроси ИИ реализовать функцию, затем перепиши её лучше».
- **Предвзятость алгоритмов (Bias).** Необходимо обсуждать кейсы, когда ИИ ошибается чаще для определенных групп людей из-за несбалансированного датасета. Это учит учащихся тому, что код — это не нейтральная математика, а отражение данных, на которых его учили.

- **Академическая честность.** Где граница между помощью и списыванием? Вузы уже сейчас разрабатывают политики: использование ИИ для генерации идей разрешено, для написания всего кода — запрещено без явного указания [3].

5. Инструментарий для современного занятия по информатики

Для реализации вышеописанных подходов преподавателю необходим набор проверенных инструментов:

1. **GigaChat / ChatGPT / YandexGPT:** Для генерации задач, объяснения сложных концепций (например, «Объясни рекурсию на примере матрешки») и проверки эссе.
2. **GitHub Copilot:** Для демонстрации написания промышленного кода в IDE (VS Code).
3. **Google Teachable Machine:** Для быстрых экспериментов с ML без единой строчки кода.
4. **Kaggle:** Платформа с открытыми датасетами для учебных проектов по Data Science.
5. **Stepik и Яндекс.Практикум:** Платформы с элементами адаптивного обучения, где ИИ подстраивает сложность задач под уровень ученика [4].

Интеграция ИИ в дисциплину «Информатика» — это не угроза профессии преподавателя, а эволюция самой дисциплины. Мы переходим от обучения «ремеслу кодера» к обучению «искусству архитектора систем». Преподаватель становится проводником в мир, где знания устаревают за 2-3 года, а ключевой компетенцией является умение учиться и адаптироваться. ИИ в данном контексте — не конкурент человеку, а экзоскелет для его интеллекта. Воспитать поколение, которое не просто использует нейросети для генерации картинок, а понимает их архитектуру, ограничения и этические риски — стратегическая задача современного процесса обучения.

Библиографический список:

1. Иванова, А. В. Искусственный интеллект в школьном курсе информатики: новые горизонты обучения / А. В. Иванова // Сборник материалов конференции «Цифровая педагогика». — 2023. — С. 45–58.
2. Петров, С. С. Адаптивные обучающие системы на базе ИИ: теория и практика / С. С. Петров // Информатика и образование. — 2022. — № 5. — С. 12–21.
3. Бурцева, И. В. Применение искусственного интеллекта на уроках информатики для генерации задач / И. В. Бурцева // Педагогический опыт. — 2024. — № 2. — С. 110–115.
4. Смирнов, К. Л. Методика обучения основам искусственного интеллекта в школьном курсе / К. Л. Смирнов // Инфоурок: методические материалы. — 2023. — URL: https://infourok.ru/metodika-obucheniya-osnovam-iskusstvennogo-intellekta-v-shkolnom-kurse-informatiki-7431419.html (дата обращения: 23.10.2026).