



УДК 004.8

DOI: <http://dx.doi.org/10.21686/1818-4243-2024-4-23-32>

И.А. Яшина

Красноярский государственный педагогический университет
им. В.П. Астафьева, Красноярск, Россия

Искусственный интеллект в обучении программированию студентов педагогического вуза

Цель исследования. Современное образование подвергается значительным изменениям. Технологии искусственного интеллекта внедряются в образовательную практику вузов, в том числе и педагогических. Данные технологии имеют большие возможности. Однако необходимо более глубокое изучение влияния и роли искусственного интеллекта не только в образовательном процессе вуза в целом, но и в специфике преподавания отдельных дисциплин, включая программирование для будущих учителей информатики. Проблема использования искусственного интеллекта в обучении студентов педагогических вузов является актуальной. Статья посвящена обоснованию роли искусственного интеллекта в процессе обучения программированию студентов, будущих учителей информатики и исследованию возможности его результативного применения в реальном учебном процессе.

Материалы и методы. Выполнен анализ научных публикаций в области использования искусственного интеллекта в образовательном процессе вуза и анализ научно-методической литературы по обучению программированию студентов педагогических вузов. Проанализированы средства искусственного интеллекта, используемые в профессиональной разработке программного обеспечения. Проведен эксперимент по использованию в курсе программирования на Python для студентов педагогического вуза онлайн среды разработки Replit с технологией искусственного интеллекта. Выполнено анкетирование студентов в начале и после завершения учебного курса.

Результаты исследования. Традиционно в обучении программированию студентов 1 курса Красноярского государственного педагогического университета используется авторский электронный учебный курс, разработанный в среде Moodle, тренажёр по программированию с автоматизированной проверкой решений, IDLE Python и онлайн компиляторы. Данные средства обучения

были дополнены онлайн средой разработки Replit с технологией искусственного интеллекта. По результатам эксперимента показано, что использование онлайн среды разработки Replit с технологией искусственного интеллекта при обучении программированию студентов педагогического вуза помогает в написании и отладке кода, делает курс менее сложным, это подтверждается результатами анкетирования студентов, также показывающими, что большинство студентов отдают свое предпочтение данной среде разработки. По окончании освоения студентами курса программирования отмечается значительный рост в самооценке уровня программирования, что подтверждается учебными результатами.

Заключение. Роль искусственного интеллекта в процессе обучения студентов программированию может быть значительна и полезна как для преподавателя, так и для обучаемых. В частности, использование искусственного интеллекта помогает студенту в написании и отладке кода, упрощает процесс объяснения преподавателем способов разработки алгоритма и составления программы. На примере интегрированной онлайн-среды Replit, имеющей встроенный искусственный интеллект, показаны ее возможности для результативного обучения студентов программированию.

В результате статьи подтверждается необходимость дальнейшего исследования и разработки методического обеспечения для успешного интегрирования искусственного интеллекта в процесс обучения студентов педагогических вузов в области программирования.

Ключевые слова: искусственный интеллект в обучении программированию, среда разработки Replit, методика обучения программированию, искусственный интеллект в программировании.

Irina A. Yashina

Krasnoyarsk State Pedagogical University named after V.P. Astafyev, Krasnoyarsk, Russia

Artificial Intelligence in Teaching Programming to Students of Pedagogical University

The purpose of the study. Modern education is undergoing significant changes. Artificial intelligence technologies are being introduced into the educational practice of universities, including pedagogical universities. These technologies have great opportunities. However, there is a need for a deeper study of the influence and role of artificial intelligence not only in the educational process of the university as a whole, but also in the specifics of teaching individual disciplines, including programming for future teachers of computer science. The problem of using artificial intelligence in teaching students of pedagogical universities is urgent. The article is devoted to the substantiation of the role of artificial intelligence in the process of teaching programming to students, future teachers of computer science and the study of the possibility of its effective application in the real educational process.

Materials and methods. The analysis of scientific publications in the field of using artificial intelligence in the educational process of higher education institution and the analysis of scientific and

methodological literature on teaching programming to students of pedagogical universities were performed. Artificial intelligence tools used in professional software development have been analyzed. The experiment of using the online development environment Replit with artificial intelligence technology in Python programming course for students of pedagogical university was carried out. A questionnaire survey of students at the beginning and after the completion of the course was carried out.

The results of the study. Traditionally, the author's e-learning course developed in the Moodle environment, programming simulator with automated solution checking, IDLE Python and online compilers are used in teaching programming to 1st year students of Krasnoyarsk State Pedagogical University. These learning tools were supplemented with Replit online development environment with artificial intelligence technology. According to the results of the experiment it is shown that the use of online development environment Replit

with artificial intelligence technology in teaching programming to students of pedagogical university helps in writing and debugging code, makes the course less complex, this is confirmed by the results of questionnaire survey of students, also showing that most students give their preference to this development environment. At the end of the students' mastering of the programming course, there is a significant growth in self-assessment of the programming level, which is confirmed by the learning results.

Conclusion. The role of artificial intelligence in the process of teaching students programming can be significant and useful for both the instructor and the students. In particular, the use of artificial intelligence helps the student in writing and debugging code, simpli-

fies the process of explaining to the instructor the ways of algorithm development and program compilation. The example of the integrated online environment Replit, which has built-in artificial intelligence, shows its possibilities for effective training of students in programming. As a result, the article confirms the need for further research and development of methodological support for the successful integration of artificial intelligence in the process of teaching students of pedagogical universities in the field of programming.

Keywords: artificial intelligence in teaching programming, Replit development environment, programming teaching methodology, artificial intelligence in programming.

Введение

Стремительное развитие информационно-коммуникационных технологий трансформирует многие сферы жизни современного человека. Образование также подвергается значительным изменениям, это обусловлено тем, что современные дети сильно отличаются от предыдущих поколений обучающихся, большинство из них к школьному возрасту уже свободно обращаются с цифровыми технологиями. Актуальность данной работы обусловлена развитием технологий искусственного интеллекта (далее по тексту ИИ) и внедрением их в сферу образования, в том числе в подготовку будущих учителей. Задача данного исследования – обоснование роли искусственного интеллекта в процессе обучения программированию студентов, будущих учителей информатики и исследование возможности его результативного применения в учебном процессе.

Цель применения искусственного интеллекта в вузах заключается в улучшении педагогического процесса и обеспечении подготовки специалистов, обладающих не только профессиональными, но и цифровыми навыками. В отечественной практике есть примеры эффективного использования технологий искусственного интеллекта в образовательном процессе вуза. Так в Томском государственном университете (ТГУ), внедрены курсы, основанные на искус-

ственном интеллекте, однако из-за ограниченного финансирования не все из них в настоящее время поддерживаются. ИИ-платформы, разработанные в ТГУ, и их зарубежные конкуренты, такие как Squitrel (обучение математике), Voxy (обучение английскому языку) и другие, зарекомендовали себя как методические инструменты для адаптивного и персонализированного обучения [1].

Анализ публикаций по данной проблеме показывает необходимость выявления практической роли и функций искусственного интеллекта в высшем педагогическом образовании.

Согласимся с авторским коллективом Омского государственного педагогического университета, что «несмотря на то, что технологии искусственного интеллекта только начинают внедряться в современную образовательную практику на уровне вспомогательных средств, уже сейчас можно говорить о том, что их возможности достаточно широки и будут раскрыты в будущем, поэтому необходимо осмысление роли данных технологий в образовательной деятельности педагогического вуза» [2]. Авторами выделен ряд свойств искусственного интеллекта, которые они отнесли к его сильным сторонам и возможностям использования в образовательном процессе педагогического вуза. Вот некоторые из них:

– Улучшение администрирования образовательного процесса

– Индивидуализация процесса обучения

– Объективное и своевременное оценивание и контроль результатов студентов

– Глобальный доступ к образованию

Представленному списку сильных сторон и возможностей искусственного интеллекта в образовательном процессе авторы противопоставляют его свойства, относящиеся к слабым сторонам и рискам использования. Представим некоторые из них, которые, на наш взгляд, могут проявляться при внедрении искусственного интеллекта в учебный процесс отдельных дисциплин:

– Недостаточное научное и методическое обеспечение использования искусственного интеллекта в образовании

– Проблемы организации и реализации воспитания

– Высокая стоимость

Несмотря на выявленные недостатки и потенциальные риски применения искусственного интеллекта в образовательном процессе педагогических вузов, в настоящее время наблюдается его постепенная интеграция в образовательный процесс на уровне отдельных компонентов.

Работа коллег из Томского педагогического университета, посвященная практическим аспектам включения искусственного интеллекта в образовательный процесс педагогического вуза, раскрывает образовательные функции нейронных сетей (одного из видов искусственного интеллекта) и затрагивает вопросы

аватаров и антропоморфных ассистентов. Рассмотрим две функции, наиболее интересные, на наш взгляд, в обучении будущих педагогов. Авторами выделена коммуникационная функция, благодаря которой искусственный интеллект может упростить процесс объяснения материала преподавателем, стать посредником в общении или оптимизировать процесс подготовки преподавателем ряда описаний по заданиям различной сложности для обучающихся. Авторы отмечают, что «функция обучения с использованием средств ИИ внедряется, например, в японских школах, где не имитирующий внешность человека мобильный андроид проводит с детьми занятия по изучению иностранного языка. В России имеются попытки создания виртуального учителя-логопеда и преподавателя-программиста на основе ИИ» [3].

Согласимся с мнением авторов, что «в то время как роль технологий ИИ в жизни людей возрастает, будущие учителя информатики могут быть включены в процесс внедрения ИИ в образование, находя ему применение в данной сфере, используя технологии ИИ в своих проектах, вызывая интерес к перспективным технологиям у детей» [4].

Обучение программированию одна из актуальных проблем в подготовке будущих учителей информатики, подтверждением чего являются работы по выявлению условий эффективного обучения и усовершенствованию методики обучения на основе различных подходов [4-7]. Содержание и методы обучения программированию должны соответствовать современному состоянию языков, методов и технологий программирования и перспективам их развития. Цель данного исследования заключается в обосновании роли искусственного интеллекта в процессе обучения программированию

бакалавров педагогического образования по направлению «Математика и информатика» и исследовании возможности его результативного применения в учебном процессе.

Искусственный интеллект в программировании

В настоящее время искусственный интеллект становится неотъемлемой частью многих областей, в том числе и программирования. Использование ИИ в разработке программного обеспечения открывает перед программистами уникальные возможности для улучшения процессов создания, оптимизации кода, автоматизации задач и даже создания инновационных решений. ИИ не просто новый инструмент в арсенале разработчика, это полноценный активный участник в процессе создания и оптимизации программного кода [8].

Помимо генерации кода ИИ, как правило может найти ошибки в коде и объяснить сгенерированный или предложенный ему код. Такие возможности ИИ и его активное использование в профессиональной разработке программного обеспечения приводит к мыслям об использовании ИИ при обучении программированию.

Основной навык написания кода приобретается студентами за счет самостоятельной работы. Безусловно в этом процессе могут возникать затруднения, которые можно пытаться решить самостоятельно или с чьей-либо помощью. Оставаясь один на один с проблемой, не каждый студент способен справиться и не каждый может получить своевременную помощь. В такой ситуации очень нужен помощник, который бы объяснил код, или указал на ошибки в коде. Именно таким помощником может выступить искусственный интеллект.

А.П. Сазонов в своей статье [8] делает обзор инстру-

ментов, позволяющих использовать возможности ИИ для написания кода и анализирует возможности некоторых из них. Все рассмотренные им инструменты: OpenAI Codex, Tabnine, Codiga AI, GitHub Copilot, Copilot для Xcode не подходят для использования в обучении программированию студентов педагогического вуза по критериям доступности и интеграции, так как являются платными, либо предполагают использование платной среды разработки.

Следует отметить, что сегодня список ИИ инструментов, автоматизирующих задачи разработки программного обеспечения, такие как генерация кода, редактирование кода и обнаружение ошибок значительно расширен. На сайте [Waildworld](#) представлена библиотека ИИ инструментов для различных нужд, в том числе и программирования. Каталог «Кодинг» содержит 256 инструментов, не все из представленных относятся к программированию, но безусловно, там можно найти бесплатные решения с доступной интеграцией, такие как расширения браузеров или инструменты совместимые с определенными средами разработки. Особое внимание, на наш взгляд, заслуживает такое решение, как Replit – браузерная IDE для совместной работы, позволяющая писать код с помощью искусственного интеллекта.

Основанная в 2016 году, Replit похожа на Google Docs для разработчиков, позволяя им писать и совместно работать над программным обеспечением в режиме реального времени. В феврале 2023 года компания запустила бота для программистов похожего на ChatGPT (генеративная нейронная сеть Generative Pretrained Transformers).

Некоторые особенности Replit [9]:

– Не требует установки. Можно войти в систему и сра-

зу начать писать код. Можно работать на смартфоне.

– Многопользовательская работа. Позволяет работать с кем угодно в любое время, просто поделившись ссылкой. Можно видеть курсоры, правки и результаты работы друга друга в режиме реального времени.

– Опции базы данных. Предлагает различные варианты хранения и получения данных для проектов.

– Интеграция с GitHub. Можно легко синхронизировать свои изменения между Replit и GitHub.

– Ghostwriter. Функция *Ghostwriter* на основе ИИ помогает писать код быстрее и умнее. Ghostwriter может автозаполнять ваш код, предлагать улучшения, генерировать шаблонный код и даже писать целые функции. Ghostwriter работает с языками Python и JavaScript

Рассмотрим подробнее основные функции *ИИ-ассистента в Replit*, которые могут использовать студенты при обучении программированию.

Генерация кода: выглядит это следующим образом, когда вы начинаете печатать код, ИИ анализирует уже написанный код и предлагает подходящие варианты завершения. Это может быть одна строка, а может быть несколько, например, условный оператор предлагается целиком. Студенту важно понимать подходит ему предлагаемый вариант или нет. Не всегда он может быть верным, особенно, если не обозначена решаемая задача. Если условие задачи закомментировано, то чаще всего подсказки будут верными. Рассмотрим два варианта подсказок ИИ (рис.1). Подсказка – это строка кода после курсора. Вариант 1: ИИ не знает какая задача решается и предлагает чаще всего не подходящий вариант подсказки. В данном примере был предложен вариант по последней выполняемой про-

Вариант 1:

```
1 count = 0
2 number = int(input("Введите число (0 для завершения): "))
3 while number != 0:
4     if number % 4 == 0 and str(number)[0] == '4':
5         count += 1
6     number = int(input("Введите число (0 для завершения):"))
```

Вариант 2:

```
1 # Найти количество четных чисел в последовательности,
  завершающей числом 0.
2 count = 0
3 number = int(input("Введите число (0 для завершения): "))
4 while number != 0:
5     if number % 2 == 0:
6         count += 1
7     number = int(input("Введите следующее число (0 для
  завершения): "))
```

Рис.1. Варианты подсказок ИИ при написании кода

Fig. 1. Variants of artificial intelligence hints when writing a code

грамме. Вариант 2: Условие задачи закомментировано. В этом случае, предлагаемый вариант подсказки, чаще всего будет правильным. После заголовка цикла, он предложит условный оператор. Затем, если правильно ставить курсор будут предложены последующие подсказки.

Еще один вариант генерации кода – это сделать запрос в чате. Запрос можно сформулировать так: «предложи код для задачи...» и далее формулировка задачи и какие-либо уточняющие детали. Большинство учебных задач генерируются верно, но важно, чтобы используемые средства соответствовали изученному материалу. Например, если списки еще не изучали, то их нельзя использовать в решении. После того, как код будет сгенерирован, можно вести диалог в чате, уточнять свой запрос и анализировать предлагаемый код.

Неважно как будет написан код, с использованием подсказок ИИ или нет, в любом случае может возникнуть ошибка и необходимость отладки кода. Безусловно студенты должны учиться отлаживать свои программы. Но процесс самостоятельной отладки кода может

занимать очень много времени или даже приводить в тупик. Важно, чтобы таких ситуаций было как можно меньше, особенно во время самостоятельной работы дома, т.к. это может снижать мотивацию к написанию программ.

Нахождение ошибок: в Replit у ИИ-ассистента есть функция Debug, с помощью которой он проверит код и опишет какие обнаружил ошибки и прокомментирует как их можно исправить. По умолчанию он делает это на английском языке, при желании можно попросить дать пояснения на русском. Рассмотрим код с тремя ошибками (рис. 2): отсутствие закрывающей скобки (строка 2), отсутствие двоеточия после условия в цикле и неверные отступы перед оператором присвоения (строка 6), так же на рисунке представлены пояснения ИИ-ассистента, перед которыми им была описана ошибка и приведен корректный код. Если какие-то пояснения не понятны или кажутся не точными, то их можно уточнить, задав дополнительные вопросы. Так, например, получилось с первой ошибкой: в всплывающем окне на рисунке (рис. 2) дается пояснение, что скобка не закры-

```

1 count = 0
2 number = int(input("Введите число (0 для завершения): "))
3 while number
4     if number
5         count
6         number = int(input("Введите число (0 для завершения): "))
7         print("Количество четных чисел:", count)

```

AI BASIC MODEL 7 minutes ago Context Rerun

Объяснение исправления:

- Мы разделили строку на две: в первой строке мы получаем ввод пользователя и присваиваем его переменной `number`, а во второй строке начинаем цикл `while`.
- После условия `while number != 0` мы добавили двоеточие (`:`) - это необходимо, чтобы обозначить начало блока кода, который будет выполняться внутри цикла.
- Внутри цикла `while` мы отступили вправо - это необходимо, чтобы указать Python, что эти строки кода относятся к циклу.

Теперь код работает правильно: он запрашивает у пользователя число, проверяет, четное ли оно, и увеличивает счетчик `count`, если это так. Цикл будет повторяться до тех пор, пока пользователь не введет число 0. После завершения цикла, программа выводит количество четных чисел.

Рис. 2. Исправление ошибок в коде с помощью ИИ

Fig. 2. Correcting errors in code with the help of artificial intelligence

1. Инициализация:

- `count = 0`: Мы создаем переменную `count` для хранения количества четных чисел. Сначала она равна 0, потому что мы еще не нашли ни одного четного числа.
- `number = int(input("Введите число (0 для завершения): "))`: Код просит пользователя ввести число. `input()` запрашивает ввод с клавиатуры, а `int()` преобразует его в целое число. Это число записывается в переменную `number`.

2. Цикл и подсчет:

- `while number != 0`: Это начало цикла. Он будет выполняться до тех пор, пока пользователь не введет 0.
- `if number % 2 == 0`: Внутри цикла код проверяет, является ли введенное число четным. `%` - это оператор остатка от деления. Если остаток от деления `number` на 2 равен 0, то число четное.
- `count += 1`: Если число четное, значение `count` увеличивается на 1.
- `number = int(input("Введите число (0 для завершения): "))`: После каждой проверки код снова просит пользователя ввести число, чтобы цикл мог продолжить работу.

3. Вывод:

↓ Scroll to latest

Рис. 3. Фрагменты объяснения искусственным интеллектом кода

Fig. 3. Fragments of artificial intelligence explanation of the code

та, а ИИ выдал ошибку синтаксиса `SyntaxError: Expected ',', found 'while'`. Но, указав ему на это, он дает вполне разумное объяснение.

Объяснение кода: одна из самых полезных функций при обучении. В чате ИИ можно задать вопрос на объяснение любого фрагмента кода, а также всей программы. Кроме того, у ИИ-ассистента есть отдельная функция `Explain`, которая сразу формулирует вопрос: «How does `main.py` work?». Затем можно попросить дать ответ на русском языке или сразу переформулировать вопрос по-русски. Объяснения в этих двух вариантах могут отличаться.

На рисунке (рис. 3) представлен фрагмент объяснений ИИ программы нахождения количества четных чисел в последовательности, завершающейся числом 0.

Ввиду наличия бесплатного тарифа и, перечисленных выше свойств, Replit может быть выбран в качестве среды разработки при обучении студентов педагогического вуза программированию. Все проекты и практические работы по изучаемым темам, выполненные в Replit, можно хранить в отдельных репликах (папках), что удобно структурирует материалы. Перечисленные особенности Replit де-

лают его очень удобной средой разработки.

Использование Replit в обучении программированию

Рассмотрим опыт использования ИИ в обучении программированию студентов 1 курса Института математики, физики и информатики Красноярского государственного педагогического университета им. В.П. Астафьева.

Курс программирования на 1 курсе преподается с использованием языка Python и носит «выравнивающий» характер, т.к. большая часть студентов очень слабо подготовлены в области программирования. Можно считать, что это базовый курс программирования для начинающих. Данный курс всегда вызывает затруднения у студентов в освоении, т.к. он не продолжительный и предполагает большую долю самостоятельной работы.

Курс программирования для педагогического вуза подробно описан авторами Новосибирского государственного педагогического университета [4]. В КГПУ им. В.П. Астафьева для будущих учителей математики и информатики курс программирования имеет схожую структуру и электронную поддержку. Одним из отличий является использование внешних ресурсов.

В 2022-23 учебном году Replit использовался в учебном процессе в основном для организации самостоятельной работы и без использования ИИ-ассистента. В Replit были подготовлены шаблоны заданий для самостоятельной работы по всем темам в виде отдельных реплик. На рисунке (рис.4) представлен пример шаблона по учебному модулю 3 – Цикл While, который включает шесть заданий в виде отдельных файлов `task_1.py`–`task_6.py`. Студенту достаточно сделать копию данного шабло-

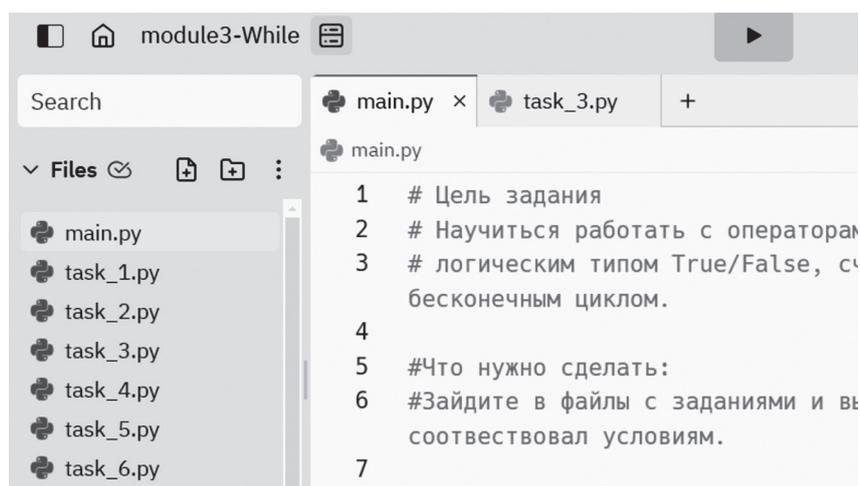


Рис.4. Пример шаблона с заданиями в Replit
Fig. 4. Example of a template with tasks in Replit

на, в ней выполнить задачи и отправить ссылку. Это сделало работу студентов удобной, т.к. работать можно из любого места и устройства, удобное хранилище выполненных работ. Для преподавателя осталась проблема проверки кода, т.к. она была в ручном режиме.

Ключевую роль в процессе обучения занимает навык разработки программ, который развивается во время практических занятий аудиторных или самостоятельных. И тут на помощь могут прийти системы обучения, которые позволяют автоматически проверять код. Такие системы могут быть двух типов: закрытые, разрабатываются отдельными вузами, как, например, CodeHedgehog в ТГУ и общедоступные, например, Питонтьютор (python-tutor.ru). Эффективность использования систем обучения программированию с автоматизированной проверкой решений показана в нескольких работах [1, 10]. Безусловно, кастомные системы обладают большим функционалом и возможностями для организации самостоятельной работы, и используются они, как правило, при обучении студентов ИТ-специальностей.

При обучении программированию студентов педагогических вузов тоже целесообразно

использование подобных систем обучения, так как они сильно упрощают работу преподавателя в плане проверки кода и позволяют получить студентам практический навык разработки программ. Поэтому в рамках самостоятельной работы студентов, кроме шаблонов Replit было решено использовать систему обучения Питонтьютор. В Питонтьюторе можно создавать классы, записывать туда студентов и видеть все выполненные ими задания. К сожалению, возможности видеть само решение у преподавателя нет. Основная цель использования данного ресурса – это тренажер, поэтому видеть, что студент выполнил задание и система его приняла достаточно, при этом необходимо контролировать, что студент может объяснить код. Всего в Питонтьюторе 11 занятий (тем) с общим числом задач 127.

Конечно, с появлением ИИ-ассистента в Replit было бы интересно использовать возможности этой технологии в обучении программированию. Однако, кроме интереса, возникли сомнения и вопросы:

– Целесообразно ли использование ИИ при обучении основам программирования?
– На каком этапе обучения лучше начать его использование?

– Будет ли ИИ помощником в освоении или его будут использовать по типу готовых решений из Интернета?

Т.к. курс программирования на Python начальный и в дальнейшем у студентов будет курс программирования на C++ и практикумы по решению задач с акцентом на решение задач школьного курса программирования и задач ЕГЭ, было решено в 2023–2024 учебном году провести эксперимент по использованию ИИ в обучении.

Эксперимент по использованию ИИ в обучении программированию

Учебный курс «Программирование» для студентов I курса начался во втором семестре. На первом занятии был проведен опрос, в нем участвовали 40 человек, который показал, что: 55% опрошенных не изучали программирование в школе, 65% не умеют записывать алгоритм решения задачи с помощью блок-схем, 77% отметили интерес к предстоящему курсу, из них, 62% считают, что им это пригодится в будущем.

Как было описано выше при обучении использовался электронный учебный курс, Replit с шаблонами заданий для самостоятельной работы и Питонтьютор в качестве тренажера задач. В качестве среды разработки использовались Replit, IDLE Python и онлайн-компиляторы. В одной группе основным инструментом разработки был Replit, в другой IDLE Python.

Использование ИИ при написании кода началось при изучении темы «Циклы», до этого ИИ в Replit был отключен или использовалась IDLE Python. Т.е. студенты сначала попробовали самостоятельно писать простые программы, немного освоились в средах разработки и затем началось постепенное

включение ИИ в процесс программирования. На начальном этапе работе с искусственным интеллектом в Replit студентов не обучали. Они увидели, что при написании кода появляются подсказки, некоторое время они их перепечатывали, затем стали разбираться как этими подсказками воспользоваться. В последствии было рассмотрено как вести диалог с ИИ и находить ошибки. Группе, у которой Replit был основным инструментом разработки не запрещалось пользоваться ИИ на лабораторных занятиях, но и не акцентировалось, что это нужно делать. Написанные программы необходимо было объяснить преподавателю. В обучении использовалась рейтинговая система оценивания, которая показала, что к концу курса, перед зачетом, количество набранных баллов студентами соответствовало традиционной пятибалльной школе в следующих пропорциях: отлично – 34% (16), хорошо – 21% (10), удовлетворительно – 32% (15) и неудовлетворительно – 13% (6). Что говорит в целом об успешном освоении курса.

По окончании курса, с целью оценки мнения студентов об электронном учебном курсе и об используемых инструментах программирования, в том числе и ИИ был проведен опрос. В опросе приняло участие 47 человек. Рассмотрим полученные результаты по вопросам, касающимся использования ИИ, предварительно проанализировав оценку уровня программирования.

Студентам было предложено оценить свой уровень программирования в начале и в завершении учебного курса (рис. 5-6).

Анализ ответов показал, что доля студентов, повысивших свой уровень программирования, в общем, возросла более чем у 90%: низкий уровень программирования в начале курса отметили 34 человека (73%), в конце курса 1 человек



Рис.5. Оценка уровня программирования в начале изучения курса
Fig. 5. Assessment of the programming level at the beginning of the course



Рис.6. Оценка уровня программирования после изучения курса
Fig. 6. Evaluation of programming level after the course study

(2%), средний уровень с 19% вырос до 45%, хороший (уверенный) с 4% до 40%, отличный и высокий уровень вначале отмечали 2 человека (4%), в конце 6 человек (13%).

50% (24) опрошенных отметили, что курс им понравился, но было сложно. Системы, в которых программировали во время обучения: Replit – 92% (43), IDLE Python – 83% (39), онлайн компиляторы – 49% (23), другое – 4% (2).

На вопрос: «Какой из систем Вы отдали бы свое предпочтение?»: 72% (34) ответили Replit, 15% (7) - IDLE Python,

9% (4) - онлайн компиляторам и 4% (2) другое.

Причину своего предпочтения указали: Удобство написания кода 81% (38), удобство отладки кода 32% (15), удобство хранения программ 62% (29), возможность использования ИИ в написании кода – 66% (33).

Следующий вопрос, который нас интересовал и был задан студентам: «Как вы считаете, использование ИИ при написании кода в Replit помогло вам?» Большая часть студентов (64% и более) отметила, что использование ИИ упрощает написание кода и помо-

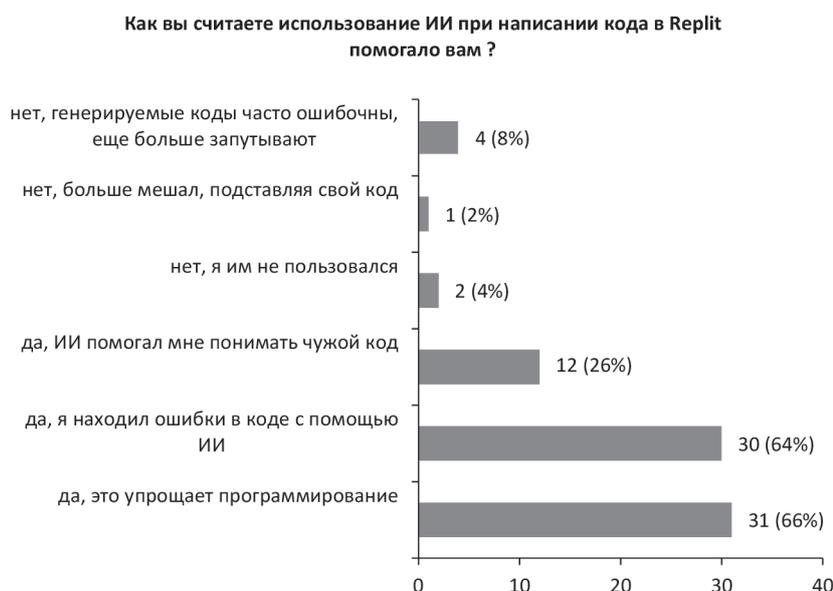


Рис. 7. Диаграмма ответов на вопрос о пользе ИИ

Fig. 7. Diagram of answers to the question about the benefits of artificial intelligence



Рис. 8. Диаграмма ответов на вопрос о использовании ИИ

Figure 8. Diagram of answers to the question about the use of artificial intelligence

Таблица (Table)

Сравнение результатов опроса в двух группах
Comparison of survey results in two groups

Основной инструмент	Replit	IDLE Python
Отдают предпочтение Replit	88 %	57 %
Привлекает возможность использования ИИ в написании кода	75 %	57 %
Привлекает возможность находить ошибки с помощью ИИ	75 %	52 %
Считают, что ИИ упрощает программирование	63 %	70 %

гает при нахождении ошибок, а 25% отметили что ИИ помогает понять чужой код (рис. 7).

Так же нас интересовало мнение студентов по поводу использования ИИ именно при обучении основам программирования: 79 % (37) отметили, что это полезно и 21 % (10) указали, что нет (рис.8).

И последний вопрос, который касался ИИ, был такой: «Если я не мог решить задачу, то, как правило, обращался за помощью к: (выберите один вариант, который использовался чаще)»: друзьям – 38 % (18), ИИ – 36 % (17), искал решение в Интернете – 23 % (11), оставлял задачу не решенной – 2 % (1). Из представленных ответов видно, что к ИИ обращались так же часто, как и к друзьям.

Если сравнить результаты по отдельным группам, то факт работы в Replit почти одинаков в обеих группах 92 % и 91 %. Этот показатель не равен 100 % т.к. сильные студенты работали по индивидуальной программе и могли выполнять работу в любой среде. Группа, в которой Replit был основным средством разработки (использовался на лабораторных) показывает больший процент в оценке пользы ИИ при написании кода.

По представленным результатам можно сделать вывод о том, что использование ИИ при обучении программированию студентов педагогического вуза помогает в написании и отладке кода, делает курс менее сложным и может заменить преподавателя, объясняя код при затруднениях с его пониманием. Интегрированную онлайн-среду разработки Replit удобно использовать в качестве основной среды разработки, т.к. она удобна в использовании, позволяет писать код с любого устройства и в любое время, позволяет делать совместные проекты, удобна для хранения и систематизации выполненных работ и имеет ИИ-ассистента. Следует отме-

титель, что Replit имеет тарифный план Teams for Education, который бесплатен для учителей и студентов, использующих при регистрации свой адрес электронной почты, связанный с образовательным учреждением. Он включает в себя неограниченное количество частных реплаев для учителей и учеников, неограниченное количество классов, неограниченное количество заданий, автоматическую оценку, инструменты обратной связи, многопользовательскую совместную работу, веб-хостинг, опции базы данных и интеграцию с Google Classroom и Clever [10]. Предполагается, что в следующем учебном году будет проведена апробация данного тарифного плана.

Заключение

В данной статье рассмотрена проблема использования искусственного интел-

лекта в обучении студентов педагогических вузов. Опыт использования интегрированной онлайн-среды разработки Replit, имеющей встроенный искусственный интеллект, подтвердил результативность ее применения в обучении программированию студентов педагогического вуза. Использование возможностей ИИ в написании, отладке и понимании кода оценены студентами положительно. Роль искусственного интеллекта в процессе обучения студентов программированию может быть значительна и полезна как для преподавателя, так и для обучаемых. В частности, использование искусственного интеллекта помогает студенту в написании и отладке кода, упрощает процесс объяснения преподавателем способов разработки алгоритма и составления программы. Данная работа является началом исследований и разра-

ботки научного обоснования и методического обеспечения использования искусственного интеллекта в образовательном процессе педагогического вуза в специфике обучения программированию. Технологии искусственного интеллекта представляются перспективными, они начинают занимать значимое место в образовании и в будущем интенсивность их использования будет только увеличиваться. Интеграция искусственного интеллекта в образовательный процесс педагогических вузов открывает новые возможности для эффективного обучения будущих учителей и развития их компетенций в области информационных технологий. Продолжение исследований и внедрение ИИ в учебный процесс педагогических вузов должно способствовать подготовке специалистов, актуальных для цифровой эры.

Литература

1. Агальцова Д.В., Валькова Ю.Е. Технологии искусственного интеллекта для преподавателя вуза [Электрон. ресурс] // МНКО. 2023. № 2(99). Режим доступа: <https://cyberleninka.ru/article/n/tehnologii-iskusstvennogo-intellekta-dlya-prepodavatelya-vuza> (Дата обращения: 29.06.2024).
2. Богдашин А.В., Соловьёв Д.Н., Соловьёва Т.О. Роль искусственного интеллекта в образовательном процессе педагогического вуза [Электрон. ресурс] // Вестник Омского государственного педагогического университета. Гуманитарные исследования. 2022. № 4(37). С. 156–160. Режим доступа: <https://cyberleninka.ru/article/n/rol-iskusstvennogo-intellekta-v-obrazovatelnom-protseesse-pedagogicheskogo-vuza> (Дата обращения: 29.06.2024).
3. Стародубцев В.А., Нерадовская О.Р. Искусственный интеллект и иммерсивные технологии в высшем педагогическом образовании // Открытое образование. 2024. № 28(2). С. 13–23. DOI: 10.21686/1818-4243-2024-2-13-23 (Дата обращения: 29.06.2024).
4. Розов К.В., Подсадников А. В. Язык программирования Python в педагогическом вузе: от основ до искусственного интеллекта [Электрон. ресурс] // Информатика и образование. 2019. № 6. С. 26–33. Режим доступа: [\[journal.ru/jour/article/view/417\]\(https://infojournal.ru/jour/article/view/417\) \(Дата обращения: 29.06.2024\).](https://info-

</div>
<div data-bbox=)

5. Баранова Е.В., Симонова И.В. Развитие алгоритмической компетенции студентов при подготовке учителей информатики в условиях цифрового образования [Электрон. ресурс] // Перспективы науки. 2019. № 8(119). С. 113–122. Режим доступа: https://www.elibrary.ru/download/elibrary_41341465_61632668.pdf (Дата обращения: 29.06.2024).
6. Круподерова Е.П., Круподерова К.Р. Обучение языкам и технологиям программирования как компонент предметной подготовки будущих учителей информатики [Электрон. ресурс] // Проблемы современного педагогического образования. 2020. № 69(3). Режим доступа: <https://cyberleninka.ru/article/n/obuchenie-yazykam-i-tehnologiyam-programmirovaniya-kak-komponent-predmetnoy-podgotovki-buduschih-uchiteley-informatiki> (Дата обращения: 29.06.2024).
7. Пугач В.И., Тюжина И.В., Макарова Е.Л. Обучение программированию будущих учителей информатики: задачи со спирально повышающейся сложностью [Электрон. ресурс] // СНВ. 2019. № 4(29). Режим доступа: <https://cyberleninka.ru/article/n/obuchenie-programmirovaniyu-buduschih-uchiteley-informatiki-zadachi-so-spiralno-povyshayuscheysya-slozhnostyu> (Дата обращения: 29.06.2024).

8. Сазонов А.П. Использование ИИ в программировании [Электрон. ресурс] // Universum: технические науки: электронный научный журнал. 2024. № 3(120). Режим доступа: <https://universum.com/ru/tech/archive/item/17010> (Дата обращения: 29.06.2024).

9. 262 лучших Кодинг ИИ инструментов. Сайт Waildworld – библиотека лучших AI-инструментов для ваших нужд [Электрон. ресурс]. Режим доступа: [https://](https://waildworld.com/ru/tag/code)

waildworld.com/ru/tag/code (Дата обращения: 29.06.2024).

10. Лучанинов Д.В., Баженов Р.И., Дмитриев А.П., Кизянов А.О. Использование автоматизированной системы обучения программированию для организации самостоятельной работы студентов [Электрон. ресурс] // Мир науки. Педагогика и психология. 2020. № 5. Режим доступа: <https://mir-nauki.com/PDF/77PDMN520.pdf> (Дата обращения: 29.06.2024).

References

1. Agal'tsova D.V., Val'kova Yu.Ye. Artificial Intelligence Technologies for a University Teacher [Internet]. MNKO. 2023; 2(99). Available from: <https://cyberleninka.ru/article/n/tehnologii-iskusstvennogo-intellekta-dlya-prepodavatelya-vuza> (cited 29.06.2024). (In Russ.)

2. Bogdashin A.V., Solov'yov D.N., Solov'yova T.O. The Role of Artificial Intelligence in the Educational Process of a Pedagogical University [Internet]. Vestnik Omskogo gosudarstvennogo pedagogicheskogo universiteta. Gumanitarnyye issledovaniya = Bulletin of Omsk State Pedagogical University. Humanitarian Research. 2022; 4(37): 156–160. Available from: <https://cyberleninka.ru/article/n/rol-iskusstvennogo-intellekta-v-obrazovatelnom-protseesse-pedagogicheskogo-vuza> (cited 29.06.2024). (In Russ.)

3. Starodubtsev V.A., Neradovskaya O.R. Artificial intelligence and immersive technologies in higher pedagogical education. Otkrytoye obrazovaniye = Open education. 2024; 28(2): 13–23. DOI: 10.21686/1818-4243-2024-2-13-23 (cited 29.06.2024). (In Russ.)

4. Rozov K.V., Podsadnikov A.V. Python programming language in a pedagogical university: from basics to artificial intelligence [Internet]. Informatika i obrazovaniye = Computer Science and Education. 2019; 6: 26–33. Available from: <https://info.infojournal.ru/jour/article/view/417> (cited 29.06.2024). (In Russ.)

5. Baranova Ye.V., Simonova I.V. Development of students' algorithmic competence in the training of computer science teachers in the context of digital education [Internet]. Perspektivy nauki = Prospects of Science. 2019; 8(119): 113–122. Available from: https://www.elibrary.ru/download/elibrary_41341465_61632668.pdf (cited 29.06.2024). (In Russ.)

6. Krupoderova Ye.P., Krupoderova K.R. Teaching programming languages and technologies as a component of subject training of future computer science teachers [Internet]. Problemy sovremennogo pedagogicheskogo obrazovaniya = Problems of modern pedagogical education. 2020; 69(3). Available from: <https://cyberleninka.ru/article/n/obuchenie-yazykam-i-tehnologiyam-programmirovaniya-kak-komponent-predmetnoy-podgotovki-buduschih-uchiteley-informatiki> (cited 29.06.2024). (In Russ.)

7. Pugach V.I., Tyuzhina I.V., Makarova Ye.L. Teaching programming to future computer science teachers: tasks with spirally increasing complexity [Internet]. SNV. 2019; 4(29). Available from: <https://cyberleninka.ru/article/n/obuchenie-programmirovaniyu-buduschih-uchiteley-informatiki-zadachi-so-spiralno-povyshayuscheysya-slozhnostyu> (cited 29.06.2024). (In Russ.)

8. Sazonov A.P. Using AI in programming [Internet]. Universum: tekhnicheskiye nauki: elektronnyy nauchnyy zhurnal = Universum: technical sciences: electronic scientific journal. 2024; 3(120). Available from: <https://universum.com/ru/tech/archive/item/17010> (cited 29.06.2024). (In Russ.)

9. 262 luchshikh Koding II instrumentov. Sayt Waildworld - biblioteka luchshikh AI-instrumentov dlya vashikh nuzhd = 262 Best Coding AI Tools. Waildworld website - a library of the best AI tools for your needs [Internet]. Available from: <https://waildworld.com/ru/tag/code> (cited 29.06.2024). (In Russ.)

10. Luchaninov D.V., Bazhenov R.I., Dimitriyev A.P., Kizyanov A.O. Using an automated programming training system to organize students' independent work [Internet]. Mir nauki. Pedagogika i psikhologiya = World of Science. Pedagogy and Psychology. 2020; 5. Available from: <https://mir-nauki.com/PDF/77PDMN520.pdf> (cited 29.06.2024). (In Russ.)

Сведения об авторе

Ирина Александровна Яшина

Красноярский государственный педагогический университет им. В.П. Астафьева,
Красноярск, Россия
Эл. почта: kulakova_do@kspu.ru

Information about the author

Irina A. Yashina

Krasnoyarsk State Pedagogical University named after V.P. Astafiev,
Krasnoyarsk, Russia
E-mail: kulakova_do@kspu.ru