Evaluating ChatGPT-driven Automated Test Generation for Personalized Programming Education

Christos Troussas, Christos Papakostas, Akrivi Krouska, Phivos Mylonas, Cleo Sgouropoulou
Department of Informatics and Computer Engineering, University of West Attica, Egaleo, Greece
Email: {ctrouss, cpapakostas, akrouska, mylonasf, csgouro}@uniwa.gr

***Abstract*— Large Language Models (LLMs), such as ChatGPT, hold immense potential to irrevocably influence the dimension of educational technology by providing empowerment for personalized learning experiences. This paper covers the integration of ChatGPT into existing eLearning platforms toward supporting Java programming education. Using artificial intelligence-based capabilities of ChatGPT in natural language processing, our software enables instructors to generate individual module assessments tailored to a student's profile with unprecedented ease. Evaluation of the effectiveness and efficiency of the system was performed by a comprehensive review of the system, obtaining feedback from instructors and students, and analyzing test performance metrics. The evaluation showed that most teachers found the system very user friendly, with significant savings in time for test creation. Satisfaction related to personalized tests designed by ChatGPT was also adequate, and the average scores achieved on test cases set by ChatGPT were relatively high compared to those manually curated. Results underline the potential of ChatGPT-driven automated test generation for enhancing personalized programming education on eLearning platforms, making available tailored assessment, by consideration of individual needs of students, and increased learning outcomes and efficiency.**

Keywords*— *Large Language Models, ChatGPT, eLearning, Programming Education, Automated Test Generation, Personalized Learning, Java Programming, Natural Language Processing, Educational Technology, Assessment

I. INTRODUCTION

In the age of digitization, eLearning has been an important element in modern education, allowing learners all over the world outstanding access and flexible learning [1]. Key to the eLearning experience is the assessment of student progression and comprehension, which has significant meaning for the realization of learning results and formulations concerning instructional strategy [2, 3]. The need for instructors to compose diverse and effective assessments is paramount, for this creates avenues to measure student understanding and know where improvement is required, thereby tailoring the teaching approach accordingly.

Personalization has increased in the last years to assume a prominent position in the enhancement of the efficiency of educational experiences [4, 5]. Tailoring assessment and learning materials to suit the needs and preferences of individual students leads to increased engagement, motivation, and knowledge retention [6, 7]. Hence, there is a growing demand for customized eLearning applications that will support personalized learning experiences by adapting to the unique learning style of each student.

Artificial Intelligence (AI) has further initiated new solutions in eLearning, especially in personalized assessments. ChatGPT [8] represents one of the Large Language Model (LLM) solutions to be at the forefront of this evolution, developed by OpenAI. Equipped with astounding capabilities in natural language understanding and generation, ChatGPT thrives on comprehending and generating human-like text. Such capabilities potentially enable ChatGPT to serve as a very good candidate for creating custom-made assessments tailored to individual student profiles, hence supporting personalized learning experiences.

This paper presents the integration of ChatGPT inside eLearning software to create personalized tests in the context of Java programming education. The latest development of the natural language processing of ChatGPT makes it possible for instructors to create fully automatic tests, with generated tests aligned according to learning objectives, expected proficiency levels, and the preference of students. The paper provides an assessment of the effectiveness of ChatGPT in improving quality and personalization for assessment in Java programming education through user studies, evaluation of performance, and system analysis.

II. RELATED WORK

AI has become part of present educational environments, helping innovatively to improve both the teaching and learning processes. Application areas of AI include intelligent tutoring systems, tailored learning environments, automatic grading, and administrative support [9-13]. In [14], the authors focus on the exploration of methods for the integration of Artificial Intelligence into informatization education. They outline relevant insights from comprehensive reviews of the uses of AI in learning settings for benefits, along with its dangers, in their proposed strategies to mitigate the risks in [15].

eLearning tools have evolved in the following years, transforming the landscape of education in a resultant time and making it more accessible as well as flexible. Now online learning platforms like Coursera, edX, and Khan Academy offer courses across disciplines using multimedia resources and interactive content to help learn. According to [16], such platforms not only democratize learning but also use among others, advanced features like peer assessment, gamification, and interactive simulations to engage learners. The integration of AI into these platforms further improves their capabilities,

as evidenced in systems that offer personalized course recommenders and automated support.

Automated test generation is indeed one of the current hot topics in the arena of "educational technology". Traditional methods of test derivation are quite labor and time-consuming, and to a greater extent, they also require too much expertise to ensure that questions are valid and reliable enough. In relation, AI and natural language processing (NLP) both have shown some potential of automating this procedure. For example, in [17 it was indicated that NLP could be used to automatically generate multiple-choice questions from text. Recent innovations, as reported in [18], address generating complex types of questions, including, for instance, fill-in-the-blank, short answer questions, or even coding exercises, automatically. Idea The systems should be able to produce many pedagogically valid questions of different types in an attempt to also relieve teachers of the associated burdens, besides providing for uniform quality in design.

The rapidly changing AI landscape, in most instances fast-tracked through ChatGPT potential, the opportunities that open become very vast. Recent literature [19-30] indicates the continuous fascination of using ChatGPT as a chatbot with a host of other roles: tutor and mentor, with context-aware support. The research is thus directed at making use of this capability of ChatGPT to improve the student's leaning experience while striving for improved performance and satisfaction levels. For example, the paper recommends that strong evaluation of the generated automated test questions be done with human oversight to carve out clear solutions to the emerging challenges in content quality and accuracy. Moreover, it highlights that reducing bias and enhancing fairness in AI-generated materials are crucial for minimizing the possibility of further deepening educational inequalities.

III. METHODOLOGY

A structured and systematic approach was followed for the integration of ChatGPT into the eLearning software to automatically generate tests. It started with a comprehensive requirements analysis phase, including interviews and detailed surveys with a targeted group of Java programming instructors and students. For example, 20 instructors from different universities and coding boot camps were interviewed to better understand the challenges and needs when creating assessments. Questions included the types of assessments employed, amongst which were multiple-choice questions, coding exercises, and short answer questions, while others were on question preferences regarding difficult levels. At the same time, some 70 students taking courses in Java programming were evaluated to determine their experiences, preferences on how they would like to be given feedback, and where more practice would be needed. This phase also included a technical review of the existing eLearning platform for its compatibility with the API of ChatGPT regarding the necessary API support and data integration capabilities.

Requirement analysis insights drove the system design phase. The design described the architecture of the system, defining all the components in the architecture: the User Interface (UI), a module for the integration of ChatGPT, the test database, and the analytics module. For instance, the instructor UI was designed with input fields for specifying test parameters like topic, difficulty, and question type. The student UI should provide a user-friendly interface for taking tests and viewing feedback. Wireframes and prototypes were created and iteratively reviewed to guarantee this interface would be satiable to a focus group of five instructors.

This step, during the integration phase, involved configuration to enable access to ChatGPT's API. This step included the processes for retrieving the appropriate API keys and setting rate limiting parameters to avoid flooding the service with requests. In this regard, a middleware application in Python was written, processing instructor inputs like "Generate a set of 10 multiple-choice questions on Java inheritance" and relaying these queries to ChatGPT. The responses provided by ChatGPT were then formatted into structured test format and stored in the system. For example, if ChatGPT wrote a question about using the "super" keyword in Java, the middleware ensured that it was properly formatted and placed under the correct category within the database.

On the other hand, a test database had been robustly designed in advance using PostgreSQL. The design of the database schema provided for different types of questions and responses for each individual student, with relevant performance metrics. For example, tables were created to hold text for the questions themselves and correct answers, metadata of the question measuring levels of difficulty, and timestamps for students' responses. Such setup guarantees independence, security, quick retrieval, and hence the best possible performance with very large datasets.

The UIs for both, the instructor and student, were developed in parallel. The instructor could enter test-related parameters via their UI, such as: the number of questions, the topics that should be covered, and how difficult it should be. For example, an instructor could select "Java loops" and request half multiple-choice and half short answer. It also gave a clean intuitive interface for the student to take tests and get feedback. Usability testing included ten instructors and thirty students, conducted in cycles to try out and base refinement of the interfaces on real user feedback.

An analytics module was developed in support of the test database to gain insights into the performance of tests and student learning outcome analytics. The module maintained algorithms on metrics such as average scores, question difficulty distribution, and time taken to complete tests. For instance, through the analytics module, reports could be generated indicating that students performed better on multiple-choice questions as compared to coding exercises (this would show areas where more instruction was needed). Used Tableau to create data visualizations that would make extensible insights more readily available to instructors.

The integrated system then underwent rigorous testing, verifying the correctness and robustness of individual components, while integration tests validated the interactions between different modules. Validation testing involved real instructors and students using the system in a controlled environment, providing feedback on usability and functionality and reporting that the feedback provided by the student UI was helpful but suggest improvements in the clarity of explanations.

Upon successful testing and validation, the system was deployed in a real educational setting for comprehensive evaluation. The effectiveness of the integration was assessed through various metrics. For instance, the quality of generated tests was evaluated by comparing them to manually created tests in terms of relevance and difficulty. User satisfaction was measured through surveys and interviews, where instructors

and students rated their experience and provided feedback. Student performance was analyzed by comparing scores on tests generated by ChatGPT with those on manually curated tests, while system performance was evaluated by monitoring server uptime, response times, and resource utilization under different loads. This comprehensive evaluation ensured that the integration met educational objectives and provided valuable insights for continuous improvement.

IV. Implementation

We proceeded to the implementation by integrating ChatGPT with eLearning software through a detailed, multi-step process focused on API configuration, integration at the backend, and database management, each of which is to be tailored in such a way as to realize a seamless user experience for both instructors and learners.

It started with the configuration of the API, where the OpenAI API keys were stored and managed in. Functions handling API requests were implemented, including setting appropriate headers and managing request limits so that valid rates of requests would not fall outside of OpenAI's usage policies. Some of these endpoints behave specifically to different types of requests to ensure optimal interaction with ChatGPT for multiple-choice questions or coding exercises.

Hence, backend integration is for setting up a server to receive the request from the User Interface and communicate effectively with the ChatGPT API. Normally, at this stage, the environment on the server is set up with Node.js for its scalability and reliability. There is a middleware application in Python that would be developed to work between these and process inputs from instructors. For instance, when questions on Java Loops are requested by an instructor, middleware models the input into an API request format compatible with ChatGPT and then interprets the response to extract and format the generated questions for integration into the platform.

Database management is critical in holding enormous data generated or to be used by the system. A PostgreSQL database is implemented to store test questions, instructor inputs, student responses, and performance data using a carefully designed schema. For example, the database schema would include tables storing question text, possible answers, correct answers, and metadata such as difficulty level and topic. The data storage solutions are optimized against the large volume of generated content so as to provide fast retrieval and secure storing of sensitive information.

It means that ChatGPT has to be plugged into the eLearning portal. This will give support to the instructor, via a systemic process, in generating tests. The instructors log in through a user-friendly interface and list the test parameters, which detail the topic, the level of difficulty, the kind of question required (multiple-choice, true/false, and coding problems) and the number of questions needed. For instance, a tutor may specify a request for "five multiple-choice questions on Java inheritance", which is then checked and pre-processed to put it in the format of a request to the API.

These inputs are taken as input by the middleware application, which forms appropriate API requests. For example, to generate a multiple-choice question about Java loops, this would involve sending a prompt like "Generate a multiple-choice question on Java for loops with four options" to the ChatGPT API. Afterward, the API response is parsed to extract the generated question and options, ensuring everything is correctly formatted and according to the specified parameters.

It then generates questions, and an instructor is allowed to review and fine-tune them through an intuitive interface. The step of reviewing these questions by instructors enables any kind of adjustments so that the questions may fit the educational standards and learning outcomes of a certain course. In this context, it also puts into practice the introduction of mechanisms for receiving feedback (it shall be reported on the quality of the questions generated by this tool), which in turn stands as the basis for improvement in the future generation of questions.

All questions generated are placed in a pool within the database from which tests may be assembled by instructors, either by selecting questions already in the pool or, where necessary, generating new ones. Features were introduced that allow randomizations of questions and answers to be made for reducing the probabilities of cheating and enhancing test variability. For instance, it would be possible for the system to shuffle the order of the questions and/or change the order of multiple-choice options each time the test is administered.

The following is a Python code snippet to demonstrate how one might use the OpenAI API for a test question generation application.

```
# Set the OpenAI API key
openai.api_key = 'the-api-key'
def generate_question(prompt):
    response = openai.Completion.create(
        engine="text-davinci-004",
        prompt=prompt,
        max_tokens=150,
        n=1,
        stop=None,
        temperature=0.7,
    )
    return response.choices[0].text.strip()


# Example usage
prompt = "Generate a multiple-choice question on Java for loops with four options."
question = generate_question(prompt)
print(question)
```

This example shows how to obtain a multiple-choice question generated for a prompt supplied by the user through the OpenAI API. The generate_question function forwards the request to the API and returns the result question.

Such eLearning software saves much time for instructors in test question preparation, which is automated. It saves instructors much valuable time that is now spent on teaching instead of administrative duties. ChatGPT can generate a wide

range of question types and difficulty levels, providing instructors with diverse options to tailor their assessments to different learning objectives. The quality control mechanism ensures that instructors can review and refine generated questions, maintaining high educational standards. Continuous improvement is facilitated by the feedback loop, where instructors' ratings and comments on generated questions help refine the output, enhancing the quality of future questions.

Here are some model test questions at a variety of levels, which ChatGPT generated for a Java programming course:

Multiple-Choice Question: Which of the following accurately develops a for loop in Java?

1. for (int i = 0; i < 10; i++)
2. for (int i = 0; i < 10)
3. for int i = 0; i < 10; i++
4. for (i = 0; i < 10; i++) Answer: 1

True/False Question: The while loop in Java always runs at least once. Answer: False

Coding Problem: A Java method is wanted, which receives an integer array and returns the sum of the elements. Answer:

```
public int sumArray(int[] arr) {
    int sum = 0;
    for (int num : arr) {
        sum += num;
    }
    return sum;
}
```

V. Evaluation and Discussion

Evaluation of the e-learning system integrating ChatGPT has considered feedback from three instructors teaching Java programming courses, and seventy postgraduate students studying them.

The opinion of the instructors was obtained through a survey and interview to determine their experience of using this system. The following questions were asked. (Table I):

- System Ease of Use: The ease of moving around all components making up the system was easy? (Rated on a scale from 1 to 5, with 5 being very easy)
- Time Savings: How much time was actually saved in creating tests compared to doing so by manual methods? (Rated on a scale from 1 to 5, with 5 indicating significant time savings)
- Customization Options: : Were you often able to find appropriate customization options in the system? (Rated on a scale from 1 to 5, with 5 indicating high satisfaction)
- Inaccuracies in Questions: Did you identify inaccuracies in the questions produced by ChatGPT? (Yes/No)
- Support for Complex Questions: Have you found the support for generating complex question types in the system to be sufficient? (Yes/No)

TABLE I. Feedback metrics - Instructors

Feedback Metric	Percentage Agreement
System Ease of Use	90%
Time Savings	85%
Customization Options	80%
Inaccuracies in Questions	60%
Support for Complex Questions	70%

Most of the instructors, teachers, and faculty members responded that they found the system very easy to use, with 90% agreeing that it was user-friendly. This high satisfaction rate was complemented by the significant time savings that were reported by 85% of instructors while creating tests via the system. It's good to note, however, that this was tempered by the fact that 60% of instructors reported that the generated questions sometimes were inaccurate, so there is certainly some room for improvement in the accuracy of the AI-generated content. Moreover, 70% would like more support for complicated question types, marking potential ways in which the capabilities of the system could be improved.

Students were given detailed questionnaires to assess their perception of the tests generated by ChatGPT, shown in Table II. The questions were:

- Variety of Question Types: Did the diversification of question types enrich the learning experience for you?
- Personalization of Tests: Did you believe that tests were very person-oriented with respect to your particular learning needs?
- Effectiveness of Tests: Do you feel tests did correctly assess your knowledge in Java programming?

TABLE II. Feedback metrics - Students

Feedback Metric	Percentage Agreement
Variety of Question Types	85%
Personalization of Tests	70%
Effectiveness of Tests	80%

Further proof of the positive effect of ChatGPT-generated tests on learning was given by student feedback. For example, a striking 85% appreciated the variation in question types, indicating enrichment in learning experiences. Besides, 70% of students reported that the suitability of tests was according to their individual learning needs, a testament to the efficacy of personalized assessments. An additional 80% found the tests to be a reality test having attested to the nature of a student's awareness of Java programming, thus making it an accurate test apparatus.

The tests performance were assessed based on the 140 postgraduate students whereby 70 students did their tests on the ChatGPT-created category and, 70 students did tests in the manually created category. The students who gave an opinion earlier took ChatGPT-created tests while the others who

offered an opinion about the tests were based on manually created tests.

The evaluation metrics included:

- Average Score: This is the average score that the students have obtained in various categories of tests.
- Standard Deviation: It implies a measure of variation in scores about the mean score..
- Time Taken: : It is the average amount of time taken by the student to complete the test.

The metrics computation involved three key areas as shown in Table III: Average score, standard deviation, and time taken.

The average score was obtained by summing up all the scores by each student in both groups, that is, students taking tests generated by ChatGPT and those taking manually curated tests. Then, the sum of the scores was divided by the total number of students in the respective groups.

The variability of the scores for each group was measured to know the standard deviation. It is calculated by knowing how much each score varies from the average score. The lower the standard deviation, the less the performance varies among students.

For the time taken, the duration that each student used to complete the test was noted. Summation of the time taken by all students in each group was done. This total time was divided by the number of students in a respective group to get the average time taken to complete the tests.

TABLE III. CALCULATION OF METRICS

Test Performance Metric	ChatGPT-generated Tests	Manually Curated Tests
Average Score	85%	70%
Standard Deviation	10 points	15 points
Time Taken	40 minutes	50 minutes

Students who took the ChatGPT generated tests performed better than students who took the manually curated tests. It scored an average of 85% for ChatGPT tests, outperforming the 70% scored on manually curated tests. This returned a significant 15% performance boost. Moreover, this standard deviation of 10 points for ChatGPT-generated tests is lower compared to 15 points scored on manually curated tests that describe less variability in performance among students, thus standing for the consistency and reliability of AI-generated assessments. It should also be mentioned that, on average, students spent 40 minutes answering the tests generated by ChatGPT, which is 20% less than what students used in answering manually curated tests, thus evidencing the efficiency and effectiveness of the automated test generation process.

In this research, the performance of the system was tested for one month under different loads. It ranges from 100 to 1000 simulated concurrent user sessions. The measurement variables of performance used in this study are system reliability, average response time, and resource utilization.

The system reliability was recorded as a percentage of uptime over time for the period under consideration, amounting to the reliability of 99.9 percent available time. The average response time would be obtained by averaging the time the system takes in answering user requests, which ranged around approximately 300 milliseconds. Resource utilization is based on the percentage of resources that the system consumes during peak usage periods, with a view that the resource utilization remained below 50 percent.

Table IV summarizes the results of the system performance evaluation:

TABLE IV. SYSTEM PERFORMANCE EVALUATION

System Performance Metric	Result
System Reliability	99.9% uptime
Average Response Time	300 ms
Resource Utilization	Below 50%

System performance evaluation showed that the eLearning platform with ChatGPT demonstrated robust reliability, fast response times, and efficient resource usage. The system reliability was 99.9% uptime, and an average response time of 300 milliseconds conferred very high availability and responsiveness of the platform. Resource utilization stayed below 50%, even under peak usage periods, emphasizing overall scalability and efficiency in regard to system handling of varied loads.

In summary, the results of the evaluation exercise presage some potential for enhancing instructor productivity and student learning with a ChatGPT integration. While these are initial promising results, it is important to deal with inaccuracies in question generation and to advance support for the more complex question types in order to optimize the exploitation of the capabilities of the system and the maximal effectiveness of its application in educational settings.

VI. CONCLUSIONS

This integration of ChatGPT into e-learning platforms for test question automatization is huge in potential to bridge the possible gaps in transforming assessment practices in education. Results from the evaluation showed a large degree of satisfaction among instructors and students, proving that AI-driven assessment tools are efficient ways to enhance learning experiences. Instructors were extremely satisfied with the system since it saves much time with its user-friendly usage; there were some accuracies noted in question generation and limited support for complicated question types. Nevertheless, students have realized that more diversified tests could be designed and also valued the tailoring and the capability of the system to quite accurately determine the understanding of Java programming concepts.

Moreover, better student performance with ChatGPT-generated tests is evidenced by higher average scores, reduced variability, and reduced completion times, which speaks for itself in terms of its efficacy as an automated test generation tool that helps enable efficient and effective assessments. Therefore, the results show that ChatGPT-based e-learning systems are promising in terms of individual education, and maybe reshaping assessment practices will be towards better improvements in outcomes at larger scales. This clearly provides an impetus for further research and development in fine-tuning the capabilities of AI-driven assessment tools more generally.

Future efforts may focus on inducing more accuracy into question generation, rich support for complex question types, and adaptive testing methods to make assessments tailored for the profile of every learner. Even more interestingly, its integration of ChatGPT-generated assessments with learning analytics platforms may turn out very instrumental in gaining insight into how students learn best and their interest areas so that timely personal feedback and intervention strategies can be designed. In the same vein, addressing these shortcomings may involve future work in these directions so as to advance capabilities and effectiveness toward AI-driven assessment tools that can finally offer more tailored, adaptive, and leading learning experiences in the digital age.

REFERENCES

[1] C. Hagedorn, D. Sauer, J. Graichen and C. Meinel, "Using Randomized Controlled Trials in eLearning: How to Add Content A/B Tests to a MOOC Environment," 2023 IEEE Global Engineering Education Conference (EDUCON), Kuwait, Kuwait, 2023, pp. 1-10, doi: 10.1109/EDUCON54358.2023.10125270.

[2] K. Chrysafiadi, C. Troussas and M. Virvou, "A Framework for Creating Automated Online Adaptive Tests Using Multiple-Criteria Decision Analysis," 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Miyazaki, Japan, 2018, pp. 226-231, doi: 10.1109/SMC.2018.00049.

[3] A. Krouska, C. Troussas, and C. Sgouropoulou, "A Personalized Brain-Based Quiz Game for Improving Students' Cognitive Functions," in *Brain Function Assessment in Learning*, P. and V. P. Frasson Claude and Bamidis, Ed., Cham: Springer International Publishing, 2020, pp. 102–106. https://doi.org/10.1007/978-3-030-60735-7_11

[4] L. Zemko, K. Jelemenska, P. Cicak and T. Mifkovic, "Personalized e-Coaching as a Support in the Social Inclusion Process," 2022 20th International Conference on Emerging eLearning Technologies and Applications (ICETA), Stary Smokovec, Slovakia, 2022, pp. 679-684, doi: 10.1109/ICETA57911.2022.9974954.

[5] C. Papakostas, C. Troussas, A. Krouska, and C. Sgouropoulou, "Personalization of the Learning Path within an Augmented Reality Spatial Ability Training Application Based on Fuzzy Weights," *Sensors*, vol. 22, no. 18, 2022, doi: 10.3390/s22187059

[6] Y. Park, "Personalized Practice Exam Recommendation for Helping Students Prepare for Course Assessment," 2022 International Conference on Advanced Learning Technologies (ICALT), Bucharest, Romania, 2022, pp. 390-391, doi: 10.1109/ICALT55010.2022.00120.

[7] K. Chrysafiadi, C. Troussas, and M. Virvou, "Combination of fuzzy and cognitive theories for adaptive e-assessment," *Expert Syst Appl*, vol. 161, p. 113614, 2020, doi: https://doi.org/10.1016/j.eswa.2020.113614.

[8] "ChatGPT", OpenAI, https://chat.openai.com/chat (accessed Apr. 19, 2024).

[9] Y. Ren and L. Lan, "Application and Development Prospect of Artificial Intelligence in Quality Education," 2021 3rd International Conference on Internet Technology and Educational Informization (ITEI), Guangzhou, China, 2021, pp. 172-175, doi: 10.1109/ITEI55021.2021.00047. keywords: {Education;Focusing;Learning (artificial intelligence);Big Data;Market research;Data systems;Internet;Artificial intelligence;education;mental health;big data;intelligent education management system},

[10] X. Li, Z. Ma, Y. Tu and Y. Du, "Study on the Application of Artificial Intelligence Technology in Empowering Education : Taking "Intelligent Learning Partner" as an Example," 2021 2nd International Conference on Information Science and Education (ICISE-IE), Chongqing, China, 2021, pp. 1449-1453, doi: 10.1109/ICISE-IE53922.2021.00323. keywords: {Analytical models;Technological innovation;Information science;Education;Learning (artificial intelligence);IEEE educational products;History;artificial intelligence;education;technology;"Smart Learning Partner"},

[11] I. Vostroknutov, S. Grigoriev and L. Surat, "Search for a New Paradigm of Education and Artificial Intelligence. Place and Role of Artificial Intelligence in the New Education System," 2021 1st International Conference on Technology Enhanced Learning in Higher Education (TELE), Lipetsk, Russia, 2021, pp. 80-82, doi: 10.1109/TELE52840.2021.9482486. keywords: {Training;Education;Market research;Artificial intelligence;Information technology;Paradigm of education;information technology;model of information technology application;interactive subject room;artificial intelligence},

[12] N. Wang, "Application of Artificial Intelligence and Virtual Reality Technology in the Construction of University Physical Education," 2022 3rd International Conference on Electronic Communication and Artificial Intelligence (IWECAI), Zhuhai, China, 2022, pp. 343-346, doi: 10.1109/IWECAI55315.2022.00073.

[13] T. Jiang, W. Li, J. Wang and X. Wang, "Using Artificial Intelligence-based Online Translation Website to improve the Health Education in International Students," 2021 2nd International Conference on Artificial Intelligence and Education (ICAIE), Dali, China, 2021, pp. 25-28, doi: 10.1109/ICAIE53562.2021.00012.

[14] Y. Zhufeng and J. Sitthiworachart, "Research on Informatization Education Methods under the Background of Artificial Intelligence," 2021 2nd International Conference on Artificial Intelligence and Education (ICAIE), Dali, China, 2021, pp. 486-489, doi: 10.1109/ICAIE53562.2021.00107.

[15] Z. Qian, "Applications, Risks and Countermeasures of Artificial Intelligence in Education," 2021 2nd International Conference on Artificial Intelligence and Education (ICAIE), Dali, China, 2021, pp. 89-92, doi: 10.1109/ICAIE53562.2021.00026.

[16] Z. Qian, "Applications, Risks and Countermeasures of Artificial Intelligence in Education," 2021 2nd International Conference on Artificial Intelligence and Education (ICAIE), Dali, China, 2021, pp. 89-92, doi: 10.1109/ICAIE53562.2021.00026.

[17] D. R. CH and S. K. Saha, "Generation of Multiple-Choice Questions From Textbook Contents of School-Level Subjects," in IEEE Transactions on Learning Technologies, vol. 16, no. 1, pp. 40-52, 1 Feb. 2023, doi: 10.1109/TLT.2022.3224232.

[18] A. Hadifar, S. K. Bitew, J. Deleu, C. Develder and T. Demeester, "EduQG: A Multi-Format Multiple-Choice Dataset for the Educational Domain," in IEEE Access, vol. 11, pp. 20885-20896, 2023, doi: 10.1109/ACCESS.2023.3248790.

[19] X. Xie and S. Ding, "Opportunities, Challenges, Strategies, and Reforms for ChatGPT in Higher Education," 2023 International Conference on Educational Knowledge and Informatization (EKI), Guangzhou, China, 2023, pp. 14-18, doi: 10.1109/EKI61071.2023.00010.

[20] X. Yang, "ChatGPT Empowers the Informatization of Foreign Language Education in Colleges and Universities," 2023 International Conference on Educational Knowledge and Informatization (EKI), Guangzhou, China, 2023, pp. 39-42, doi: 10.1109/EKI61071.2023.00016.

[21] Y. Lei and C. Qi, "ChatGPT in Education: Angel or Evil? -A Conceptual Model to Explore Educator's Attitude Change toward ChatGPT," 2023 5th International Workshop on Artificial Intelligence and Education (WAIE), Tokyo, Japan, 2023, pp. 65-69, doi: 10.1109/WAIE60568.2023.00019.

[22] M. Neumann, M. Rauschenberger and E. -M. Schön, ""We Need To Talk About ChatGPT": The Future of AI and Higher Education," 2023 IEEE/ACM 5th International Workshop on Software Engineering Education for the Next Generation (SEENG), Melbourne, Australia, 2023, pp. 29-32, doi: 10.1109/SEENG59157.2023.00010.

[23] G. Li, "Personalized International Chinese Education based on ChatGPT," 2023 13th International Conference on Information Technology in Medicine and Education (ITME), Wuyishan, China, 2023, pp. 63-67, doi: 10.1109/ITME60234.2023.00024.

[24] Z. Yinping and Z. Yongxin, "Research on ChatGPT's Strategy to Promote the Digital Transformation of Education," 2023 26th ACIS International Winter Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD-Winter), Taiyuan, China, 2023, pp. 28-31, doi: 10.1109/SNPD-Winter57765.2023.10223978.

[25] Z. Tang and Y. Zhang, "Application of Generative Artificial Intelligence in English Education: Taking ChatGPT System as An Example," 2023 3rd International Conference on Educational Technology (ICET), Xi'an, China, 2023, pp. 42-46, doi: 10.1109/ICET59358.2023.10424297.

[26] A. Arista, L. Shuib and M. A. Ismail, "An Overview chatGPT in Higher Education in Indonesia and Malaysia," 2023 International Conference on Informatics, Multimedia, Cyber and Informations System

(ICIMCIS), Jakarta Selatan, Indonesia, 2023, pp. 273-277, doi: 10.1109/ICIMCIS60089.2023.10349053.

[27] Y. Li, J. Xu, Y. Zhu, H. Liu and P. Liu, "The Impact of ChatGPT on Software Engineering Education: A Quick Peek," 2023 10th International Conference on Dependable Systems and Their Applications (DSA), Tokyo, Japan, 2023, pp. 595-596, doi: 10.1109/DSA59317.2023.00087.

[28] S. Salloum, K. Shaalan, R. Alfaisal, A. Salloum and T. Gaber, "Integrating ChatGPT into Medical Education: A Combined SEM-ML Approach," 2024 International Conference on Advancements in Smart, Secure and Intelligent Computing (ASSIC), Bhubaneswar, India, 2024, pp. 1-5, doi: 10.1109/ASSIC60049.2024.10507994.

[29] V. V. D. Prasad Kotni, D. P. Ungarala, P. P. Peri, S. Nitcharla, K. V. Nagaraj and M. Mustafa, "Adoption of ChatGPT in Higher Education-Application of IDT Model, Testing and Validation," 2023 3rd International Conference on Technological Advancements in Computational Sciences (ICTACS), Tashkent, Uzbekistan, 2023, pp. 1423-1427, doi: 10.1109/ICTACS59847.2023.10390323.

[30] "OpenAI API", https://platform.openai.com/docs/api-reference (accessed Jan. 15, 2024).