

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/370596272>

ChatGPT for Education and Research: Opportunities, Threats, and Strategies

Article in Applied Sciences · May 2023

DOI: 10.3390/app13095783

CITATIONS

43

READS

1,419

2 authors:



Md. Mostafizer Rahman

The University of Aizu

34 PUBLICATIONS 370 CITATIONS

[SEE PROFILE](#)



Yutaka Watanobe

The University of Aizu

150 PUBLICATIONS 996 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Programming Education based on Deep Learning [View project](#)

Article

ChatGPT for Education and Research: Opportunities, Threats, and Strategies

Md. Mostafizer Rahman ^{1,2,*}  and Yutaka Watanobe ² ¹ Dhaka University of Engineering & Technology, Gazipur, Gazipur 1707, Bangladesh² Department of Computer Science and Engineering, The University of Aizu, Aizu-Wakamatsu 965-8580, Japan; yutaka@u-aizu.ac.jp

* Correspondence: mostafiz26@gmail.com

Abstract: In recent years, the rise of advanced artificial intelligence technologies has had a profound impact on many fields, including education and research. One such technology is ChatGPT, a powerful large language model developed by OpenAI. This technology offers exciting opportunities for students and educators, including personalized feedback, increased accessibility, interactive conversations, lesson preparation, evaluation, and new ways to teach complex concepts. However, ChatGPT poses different threats to the traditional education and research system, including the possibility of cheating on online exams, human-like text generation, diminished critical thinking skills, and difficulties in evaluating information generated by ChatGPT. This study explores the potential opportunities and threats that ChatGPT poses to overall education from the perspective of students and educators. Furthermore, for programming learning, we explore how ChatGPT helps students improve their programming skills. To demonstrate this, we conducted different coding-related experiments with ChatGPT, including code generation from problem descriptions, pseudocode generation of algorithms from texts, and code correction. The generated codes are validated with an online judge system to evaluate their accuracy. In addition, we conducted several surveys with students and teachers to find out how ChatGPT supports programming learning and teaching. Finally, we present the survey results and analysis.

Keywords: ChatGPT; educational technology; research; programming education; large language model; GPT-3; ChatGPT survey; GPT-4; artificial intelligence; AI for code



Citation: Rahman, M.M.; Watanobe, Y. ChatGPT for Education and Research: Opportunities, Threats, and Strategies. *Appl. Sci.* **2023**, *13*, 5783. <https://doi.org/10.3390/app13095783>

Academic Editor: Habib Hamam

Received: 25 March 2023

Revised: 1 May 2023

Accepted: 5 May 2023

Published: 8 May 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Over the past few years, the advancement of natural language processing (NLP) has achieved significant success in many realistic applications such as speech and entity recognition, summarization, language translation, and text generation. Researchers have also used recurrent neural network (RNN) models in many applications because of their recurrent structure to remember dependencies in texts [1–3]. However, an RNN model has limitations and cannot handle extremely long-range dependencies in natural language texts [2]. Therefore, the Transformer architecture was introduced [4] to alleviate this problem. It is an autoregressive (https://en.wikipedia.org/wiki/Autoregressive_model, accessed on 10 February 2023) and self-supervised (https://en.wikipedia.org/wiki/Self-supervised_learning, accessed on 10 February 2023) language model. Transformer also has a self-attention mechanism that determines the relationship and relevance of different parts of the input. This makes the model very robust for understanding the relationship between words in a sentence regardless of their position. Generative pretrained transformer (GPT)-3 is a large language model (LLM) based on the Transformer architecture [5,6] that has achieved significant success in NLP tasks. GPT-3 models are trained on extensive text data (approximately 175 billion trainable parameters and 570 GB of text), capable of generating human-like text and performing other language-related tasks with a high percentage of accuracy.

ChatGPT is an NLP model developed by OpenAI (<https://openai.com/>, accessed on 5 January 2023) and was launched in November 2022. ChatGPT [7] emerged as a breakthrough LLM that can generate text and maintain a human-like conversational style. As GPT-3 is trained on vast amounts of Internet data and is suitable for a wide range of downstream tasks, the ChatGPT model begins with the GPT-3 pretrained LLM. However, GPT-3 has the disadvantage of having “*poorly characterized behavior*” [8]. Therefore, to avoid toxic and untruthful outputs, ChatGPT decided to use three different strategies: (i) supervised fine-tuning, (ii) reward modeling, and (iii) reinforcement learning. The process begins by collecting a dataset of labeler demonstrations used to fine-tune GPT-3 with supervised learning. Next, a dataset of rankings of model outputs is used to further fine-tune the supervised model with reinforcement learning from human feedback. Finally, the resulting model is called InstructGPT [8]. Unlike other AI-based language models, ChatGPT generates and presents entirely new content in a real-time conversation with the user. Furthermore, ChatGPT can consistently maintain a style of dialogue that engages the user in a more realistic way, rather than providing irrelevant answers to each question. This makes ChatGPT a more unique model than other LLMs.

ChatGPT has exhibited top performance in many application domains, such as coherent content and essay generation, chatbot responses, language translation, question answering, and programming code [9,10]. In addition, research is underway to fine-tune such LLMs for specific tasks and apply transfer learning in new domains. In the context of education, both students and educators can use ChatGPT for many academic and research purposes. Educators can take advantage of ChatGPT to prepare an outline of a particular course, topic-related content for lectures, presentations on academic topics, questions, problemsets, etc. Similarly, students can be assisted by ChatGPT in solving complex problems and questions, writing essays, and explaining a specific topic to accelerate their learning [10,11]. Students can even receive programming-related support here to accelerate their learning of programming. ChatGPT has made significant progress; however, there are concerns about misuse [12–14]. Therefore, it is important to consider the potential threats (e.g., the integrity of online exams and question answering) alongside the many good applications of ChatGPT for education. Some experts have expressed concern about the future of some common practices such as programming in the era of ChatGPT [13]. Therefore, it is important to rationally evaluate the situation and prepare a suitable future educational plan in the presence of tools such as ChatGPT.

In recent decades, many technologies have emerged that have occasionally disrupted traditional practices. Therefore, people need to evaluate and consider the benefits and threats of such new technologies [10]. In the past, questions have been raised about Google and how this tool will change the way people think, read, and memorize [15]. Another educational tool is the Massive Open Online Course, which gained a significant amount of attention in early 2010, and then its performance declined because of its strategies and business models [16]. These concerns can also be applied to ChatGPT, as it has many potentials but also significant dangers. Above all, ChatGPT can be used as an educational technology in many ways, including tutors, language models, and research and teaching assistants. Furthermore, ChatGPT is distinguished from other LLMs by some of its special characteristics such as accessibility, personalization, conversational format, and cost-effectiveness. Numerous studies have been conducted to explore the application of artificial intelligence (AI) in education, including chatbots [17], programming support [18,19], language models [2,20], and NLP tools [21]. However, ChatGPT has recently been launched and is also a relatively new technology in the educational domain. To the best of our knowledge, no research has (i) conducted a comprehensive survey with students and teachers to find out how ChatGPT supports programming learning and teaching and (ii) addressed the opportunities, threats, and strategies of ChatGPT for education, research, and particularly programming education.

Aims and Contributions

In this study, we explore the opportunities, challenges, and strategies of using ChatGPT in education and research and identify strategies for potential threats. To demonstrate the effectiveness of ChatGPT in programming support, we performed coding-related experiments, such as code generation from problem descriptions, pseudocode generation of algorithms from texts, and code correction. These generated solution codes were validated using an online judge system. In addition, we conducted a comprehensive survey with students and teachers to find out how ChatGPT supports programming learning and teaching. The main contributions of this study are as follows:

- Investigated the opportunities and possible threats of using ChatGPT in educational settings, particularly in programming education
- Performed experiments with ChatGPT to illustrate how this tool can be used to support programming learning
- Conducted a comprehensive survey with students and teachers to find out how ChatGPT supports programming learning and teaching
- Presented threat-mitigation strategies in the presence of AI tools such as ChatGPT
- Discussed future educational plans and curricula in light of such revolutionary AI tools

The rest of the paper is organized as follows: Section 2 presents related work, Section 3 explores the opportunities of ChatGPT from educator and student perspectives, Section 4 presents the programming learning with ChatGPT, Section 5 presents the survey results and analysis, Section 6 explores the potential threats of ChatGPT and strategies to address them, Section 7 presents the limitations, and finally, Section 8 concludes the study.

2. Related Literature

In this section, we present recent studies on education and research using ChatGPT from the perspective of educators and learners. We also summarize published studies that use ChatGPT for different educational fields, including science, medicine, and engineering. Due to the novelty of the topic, we found few peer-reviewed scholarly papers. However, we reviewed preprints (non-peer-reviewed academic papers) in different educational branches. Muneer [22] presented the potential of AI and NLP to improve academic performance. The study employed a case study using ChatGPT, which significantly enhances academic research on economics and finance. Dowling and collaborators [23] discovered the benefits of ChatGPT for their research in finance. They also mentioned the ethical implications of this revolutionary AI tool. Rudolph et al. [24] discussed the opportunities and challenges of using ChatGPT in education. Moreover, some basic features and user interfaces of ChatGPT were presented in [24]. Furthermore, several studies [9–12,14] have published applications of ChatGPT in education.

Frieder et al. [25] tested the mathematical capabilities of ChatGPT on GHOSTS and handcrafted datasets. GHOSTS is the first collection of data expressed in natural language that was created and maintained by active mathematicians. Their experimental results proved that *“the mathematical abilities of ChatGPT are significantly below those of an average mathematics student. ChatGPT often understands the question but does not provide correct solutions.”* The performance of ChatGPT as an assistant in medical education is also significant. Kung et al. [26] evaluated the performance of ChatGPT on the US Medical Licensing Exam, which includes three exams: Step 1, Step 2CK, and Step 3. Their experimental results show that ChatGPT performed close to or near the minimum requirement on all three exams despite receiving no specific guidance or support. Gilson et al. [27] investigated the performance of ChatGPT on the medical licensing exam. These compelling results mean that ChatGPT can support medical education. A summary of articles related to medical education using ChatGPT can be found in [28].

Computer programming is a complex task requiring correct logical and syntactic implementation. AI-based models have achieved plausible success on a variety of programming tasks, including code repair, summarization, completion, correction, classification, and generation [20,29]. Alphacode [30] is a state-of-the-art LLM developed and trained

specifically to support competitive-level programming. Both ChatGPT and Alphacode perform coding-related tasks by digesting a large amount of human-generated text [31]. ChatGPT is a more general conversation engine, whereas Alphacode is more specialized for programming [31], even though these two systems use “*virtually the same architecture*” [31]. Moreover, although ChatGPT was not designed and developed for automatic code repair, this tool is still suitable for it. The performance of code debugging with ChatGPT has been presented in the study [32]. Their experimental results show that the bug-fixing performance of ChatGPT is competitive with deep learning models such as Codex and CoCoNut. Nevertheless, ChatGPT has significantly outperformed traditional code-repair approaches [32]. Jalil et al. [33] studied the performance of ChatGPT in solving software testing curriculum questions. ChatGPT could generate correct/partially correct answers and explanations in approximately 44% and 57% of cases, respectively.

Malinka and collaborators [34] studied the impact of ChatGPT on higher (university) education, especially on subjects related to computer programming. They demonstrated the effectiveness and usability of ChatGPT for handling programming assignments, exams, and homework using collected data. They also presented the misuse and benefits of ChatGPT for computer science education. In the study [35], the programming capabilities of ChatGPT for solving numerical problems were investigated. Generating code, debugging and improving code written by humans, completing code, and rewriting code in different programming languages were tested with ChatGPT for numerical algorithms. They considered several numerical problems, including the diffusion equation, Poisson equation, compressible inviscid flow, incompressible Navier–Stokes equations, solving linear systems of equations, eigenvalue problems, and storing sparse matrices. The experimental results showed that ChatGPT can successfully generate program codes for numerical problems, but there are limitations and challenges in solving problems. Moreover, ChatGPT has recently been applied to tasks in different domains such as software engineering [36], medical education [37], data augmentation [38], fifth industrial revolution [39], code generation [40], intelligent vehicles [41], and solving AI tasks [42]. In contrast, researchers are concerned about the general practices of programming in the future by ChatGPT [13]. For the convenience of researchers, we have summarized some studies using ChatGPT in Table 1.

Table 1. A list of educational and research articles about ChatGPT

Category	Article	# Citations	Description
Programming Support	Jalil et al. [33]	9	Solving questions of software-testing curriculum
	Sobania et al. [32]	34	Automatic bug fixing in code
	Qadir [10]	28	Application in engineering education
	Laurent et al. [43]	9	Support tool for HTML, CSS, and JavaScript code
	Matt Welsh [13]	12	Future of common programming practices
Medical Education and Exam	Kung et al. [26]	208	AI-assisted medical education
	Gilson et al. [27]	39	Medical education
	Antaki et al. [44]	11	Ophthalmology question answering
	Rao et al. [45]	11	Adjunct for radiologic decision-making

Table 1. Cont.

Category	Article	# Citation	Description
Finance Education and Research	Muneer [22]	18	Enhance the performance of economy and finance research
	Dowling et al. [23]	57	Research on finance
	Karsten [46]	8	Explain alternative and community finance and crowdfunding
	Adam and Demir [47]	17	Financial application
Mathematics	Simon et al. [25]	34	Presented mathematical capabilities
Law	Jonathan et al. [48]	66	Examine the performance for law school exams
Translator	Wenxiang et al. [49]	48	Performance in machine translation


3. Opportunities with ChatGPT

ChatGPT is a powerful LLM developed by OpenAI that has the potential to transform our technological interactions and lead to a significant paradigm shift. Many academic articles have been published on ChatGPT, but a review of the literature on the effects of ChatGPT revealed various viewpoints ranging from favorable to unfavorable. In mathematics and science education, calculators have become an inseparable part; similarly, ChatGPT will be an important tool for daily writing and work [50]. Sharples [51] encourages educators and learners to take advantage of the available capabilities of AI tools such as ChatGPT rather than forego their use. In this section, we present the prospects and opportunities of ChatGPT for education and research from the perspective of learners, educators, and researchers.

3.1. Opportunities for Learners

ChatGPT offers many possibilities, and this tool can be a good assistant for learners. Learners are allowed to use this tool to understand and solve complex problems. For learners who prefer experimental and hands-on learning, ChatGPT is an excellent platform to achieve this [24]. One of the biggest advantages of ChatGPT is its ability to understand and respond to natural language queries. This allows learners to ask ChatGPT a question in the same way they would ask their tutors. This makes ChatGPT more intuitive and learner-friendly. It can be used at all levels of education, from elementary to higher education, and even for professional development. The ChatGPT model can help students develop their reading and writing skills by providing suggestions (e.g., syntactic and grammatical); the model can create practice exercises and quizzes for various subjects (e.g., mathematics, physics, language, and literature); the model can provide a set of exercises and quizzes. Moreover, the ChatGPT model can create explanations and step-by-step solutions to a given problem; the model can help develop problem-solving skills and analytical and out-of-the-box thinking.

Furthermore, ChatGPT can be used for group discussions and debates by providing personalized guidance to learners during the discussion; ChatGPT can support learners with disabilities by providing services such as speech-to-text and text-to-speech. The ChatGPT model can be a professional tutor for developing language skills, programming, report writing, project management, and technical (e.g., medical, legal, and IT) report writing. More interestingly, learners can argue with ChatGPT about the given explanations, solutions, and other suggestions. Therefore, learners receive interactive help from ChatGPT anytime and anywhere. In addition, we experimented with the ChatGPT model to find the derivatives of mathematical equations, and it solved all equations correctly. In this case, the correctness is 100%. Figure 1 shows the capabilities of ChatGPT for technical education.

 Give an example of quantum physics with work flow



Sure! Here's an example of a quantum physics problem and its solution workflow:

Problem: A particle with spin-1/2 is in the state:

$$|\psi\rangle = (1/\sqrt{3})|+z\rangle - (\sqrt{2}/\sqrt{3})|-z\rangle$$

where $|+z\rangle$ and $|-z\rangle$ are the eigenstates of the z-component of the particle's spin with eigenvalues $+\hbar/2$ and $-\hbar/2$, respectively. Find the probabilities of measuring the spin of the particle along the x- and y-axes.

Solution workflow:

1. Write down the matrices for the x- and y-components of the particle's spin in the $\{|+z\rangle, |-z\rangle\}$ basis.
2. Express the state vector $|\psi\rangle$ in terms of the $\{|+x\rangle, |-x\rangle\}$ and $\{|+y\rangle, |-y\rangle\}$ bases using the appropriate rotation matrices.
3. Use the matrix representation of the x- and y-components of the spin to find the probabilities of measuring the spin along the x- and y-axes, respectively.
4. Normalize the state vector to check that the probabilities of measuring the spin along all three axes add up to 1.

The above is just a rough idea of the workflow for a quantum physics problem involving spin measurements. Each step involves the use of mathematical tools and concepts from linear algebra and quantum mechanics, such as matrix representations of operators, state vector transformations, and probability amplitudes. The specific details and complexity of each step depend on the particular problem being solved.



Find the derivative of $y=f(t)=80-4.9t^2$



To find the derivative of $y = f(t) = 80 - 4.9t^2$, we can use the power rule of differentiation, which states that the derivative of a function of the form $f(x) = x^n$ is $f'(x) = nx^{(n-1)}$.

In this case, $n = 2$, so we have:

$$f'(t) = d/dt [80 - 4.9t^2] = d/dt [80] - d/dt [4.9t^2] = 0 - 9.8t = -9.8t$$

Therefore, the derivative of $y = f(t) = 80 - 4.9t^2$ is $y' = -9.8t$.



Find the derivative of $y=f(x)=\sqrt{(169-x^2)}$



To find the derivative of $y = f(x) = \sqrt{(169-x^2)}$, we can use the chain rule of differentiation.

Let $u = 169 - x^2$, then $f(x) = \sqrt{u}$.

Using the chain rule, we have:

$$\begin{aligned} f'(x) &= d/dx [\sqrt{u}] = d/du [\sqrt{u}] * du/dx = 1/(2\sqrt{u}) * d/dx [169 - x^2] \\ &= -x/\sqrt{(169-x^2)} \end{aligned}$$

Therefore, the derivative of $y = f(x) = \sqrt{(169-x^2)}$ is:

$$y' = f'(x) = -x/\sqrt{(169-x^2)}.$$

(a)

(b)

Figure 1. Capabilities of ChatGPT for technical education. (a) Conceptual example and explanation of a topic. (b) Solutions of mathematical questions.

3.2. Opportunities for Educators

As an LLM, ChatGPT can be a valuable tool for educators in many ways. Educators can take advantage of using ChatGPT for effective teaching and research. Here are some examples that can demonstrate the effectiveness of ChatGPT for teaching and research.

Lesson Planning: ChatGPT can be used to create lesson plans for specific courses, such as math, chemistry, physics, computer science, civil engineering, language, and literature. ChatGPT provides topic-specific illustrations, activities, and exercises to help educators better teach their students. ChatGPT can also be used to generate topic-specific quiz questions tailored to the subject matter and difficulty level. For example, we asked ChatGPT to “prepare a detailed outline for the Algorithms and Data Structures course”, as depicted in Figure 2. It creates a complete table of contents for the Algorithms and Data Structures course, showing topics and a breakdown of each topic, along with learning objectives.

Personalized Learning Support: Educators can use ChatGPT to provide personalized learning support for their students. Depending on a student's needs and learning style, ChatGPT can suggest customized resources and learning activities. For instance, educators can use ChatGPT to analyze student performance data and identify areas where students are struggling with particular concepts or algorithms. An educator might notice that a particular student is struggling with sorting algorithms. In this case, the educator can take advantage of ChatGPT to generate customized resources based on that student's learning style and abilities (e.g., a video tutorial on a specific sorting algorithm that the student is struggling with or a coding exercise to reinforce that concept). Figure 3 depicts the personalized learning steps.



Prepare an detailed outline for "Algorithms and Data Structures" course in tabular form



Here's a more detailed outline for a course on Algorithms and Data Structures:

Week	Topic	Learning Objectives
1	Introduction to Algorithms and Data Structures	Define algorithms and data structures; explain their importance; analyze algorithms; use pseudo code; implement basic algorithms
2	Asymptotic Analysis	Define big-O notation and analyze algorithm time and space complexity; compare algorithms; evaluate algorithmic efficiency; perform complexity analysis
3	Arrays and Linked Lists	Implement and compare arrays and linked lists; insert, delete, and search elements; understand their trade-offs
4	Stacks and Queues	Implement and compare stacks and queues; perform basic operations; use them to solve problems; understand their applications
5	Recursion and Backtracking	Understand and implement recursive algorithms; use them to solve problems; analyze recursion; implement backtracking algorithms
6	Trees and Graphs	Implement and compare tree and graph data structures; traverse trees and graphs; perform basic operations; understand their applications
7	Sorting and Searching Algorithms	Implement and compare sorting and searching algorithms; use them to solve problems; analyze their efficiency; perform complexity analysis
8	Hash Tables and Heaps	Implement and compare hash tables and heaps; use them to solve problems; understand their trade-offs; analyze their efficiency
9	Dynamic Programming and Greedy Algorithms	Understand and implement dynamic programming and greedy algorithms; use them to solve problems; analyze their efficiency
10	Algorithm Design Techniques and Optimization	Analyze and optimize algorithms using design techniques; use data structures to solve problems; apply optimization techniques
11	Graph Algorithms: Shortest Paths and Minimum Spanning Trees	Understand and implement graph algorithms for finding shortest paths and minimum spanning trees; use them to solve problems; analyze their efficiency
12	NP-Completeness and Approximation Algorithms	Understand and analyze NP-completeness and approximation algorithms; identify problems that are NP-complete; apply approximation algorithms

Figure 2. Outline of Algorithm and Data Structures course using ChatGPT.

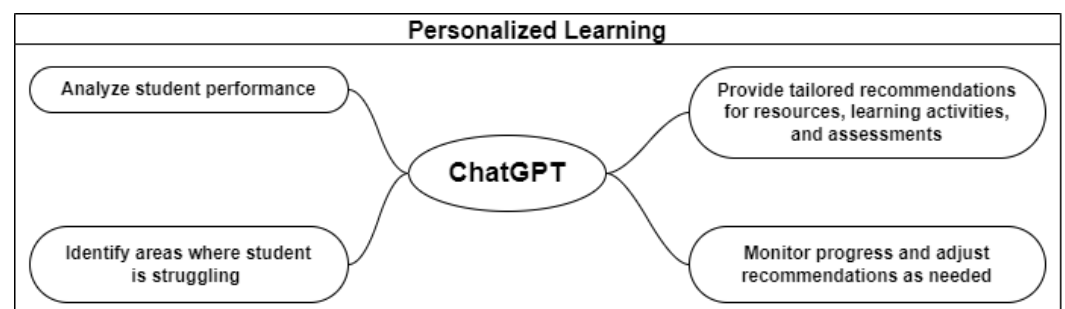


Figure 3. Personalized learning using ChatGPT.

Answering Learners' Queries: Educators can receive the help of ChatGPT to answer learners' questions. Furthermore, if educators can ask ChatGPT for explanations and examples on a particular topic, it will surely increase the effectiveness of teaching. For example, if a learner asks the question “Which sorting algorithm should we use for which data field?”, ChatGPT can provide summary information in this case (see Figure 4), which can be useful for teachers. In addition, teachers can ask ChatGPT for explanations and examples on a complex topic to obtain accurate and tailored information.



Which sort algorithm should we use for which data field?



The choice of sorting algorithm depends on several factors, including the size of the data set, the data type, and the desired sorting order (ascending or descending). Here are some general guidelines for choosing a sorting algorithm based on the data type:

1. Arrays and linked lists: Insertion sort and quicksort are usually good choices for small to medium-sized arrays, while merge sort and heapsort are more efficient for larger data sets. Insertion sort can be particularly efficient for nearly sorted data.
2. Strings: Radix sort is often used for sorting strings, as it can sort them in linear time.
3. Integers: Counting sort is a fast and efficient algorithm for sorting integers when the range of the integers is known in advance.
4. Floating-point numbers: Merge sort is a good choice for sorting floating-point numbers, as it is stable and does not suffer from rounding errors.
5. Objects and custom data types: If you need to sort custom data types, you will need to define a comparison function or operator. Once you have defined this, you can use any of the above algorithms to sort your data.

Figure 4. Summarized information generated by ChatGPT based on a query.

Rapid Assessment and Evaluation: Educators can also leverage the power of the ChatGPT model to assess and evaluate learner assignments and quizzes. The model can be used to check submitted assignments for plagiarism. Interestingly, the model can generate questions/quizzes on the basis of different difficulty levels (e.g., high, medium, easy) on the same topic. For example, as depicted in Figure 5, we asked ChatGPT to create a **high**-difficulty quiz on “*Sorting and Searching Algorithms*”. We also asked ChatGPT about quizzes of other levels, such as medium and easy, and we found that there was a significant difference between quizzes of different levels. In addition, the model can be used to grade assignments and quizzes [9]. This can save educators a significant amount of valuable time.

Apart from the above benefits, educators can use ChatGPT for language learning support, personalized feedback, professional development, and research.



Prepare a quiz on "Sorting and Searching Algorithms" with five questions and high difficulty level



Sure, here's a challenging quiz based on Sorting and Searching Algorithms:

1. Which sorting algorithm has the lowest worst-case time complexity?
 - A. Quick Sort
 - B. Merge Sort
 - C. Heap Sort
2. Which of the following is an example of an external sorting algorithm?
 - A. Quick Sort
 - B. Bubble Sort
 - C. Merge Sort
3. Which of the following searching algorithms has the worst-case time complexity of $O(1)$?
 - A. Binary Search
 - B. Linear Search
 - C. Interpolation Search
4. Which of the following sorting algorithms is most efficient for sorting a list with a small number of elements?
 - A. Quick Sort
 - B. Insertion Sort
 - C. Merge Sort
5. Which of the following statements is true about the radix sort algorithm?
 - A. It is not a stable sorting algorithm.
 - B. It has a worst-case time complexity of $O(n^2)$.
 - C. It is only applicable to integer data types.

Figure 5. ChatGPT generates quizzes with different difficulty levels on the same topic.

3.3. Opportunities for Researchers

The ChatGPT model offers many advantages to researchers. First, it can effectively support the writing process of research. At its most basic, it can improve writing by finding and correcting typographical errors, improving grammatical inconsistencies, providing advanced vocabulary, and recommending improvement strategies. This allows researchers to devote more time to experimentation and implementation. The model can also summarize published work on a particular topic, which helps researchers understand the work. It can also provide clues and research ideas by analyzing a specific topic. For example, we asked ChatGPT to provide an unexplored research idea on **"how to reduce errors related to resource constraints (time and memory limit exceeded) in code for programmers"**. Based on this query, ChatGPT provided some interesting and promising research ideas, as illustrated in Figure 6.



Provide an unexplored research idea on how to reduce resource limitation related errors (time limit exceeded, memory limit exceeded) in code for programmers



One unexplored research idea to reduce resource limitation related errors in code for programmers is to develop a hybrid approach that combines static and dynamic analysis techniques.

Static analysis involves analyzing code without executing it, while dynamic analysis involves analyzing code as it executes. By combining these techniques, it may be possible to identify potential resource limitation issues before the code is executed and also monitor the code as it runs to detect any issues that may arise.

The hybrid approach could involve using static analysis tools to analyze code and identify potential issues, such as memory leaks or infinite loops. These tools could also provide suggestions for optimizing the code, such as reducing the number of operations or using more efficient data structures.

At runtime, the code could be monitored using dynamic analysis tools to detect any resource limitation errors that may occur. This could involve tracking memory usage, execution time, and other metrics to identify potential issues in real-time. The dynamic analysis tools could also provide suggestions for optimizing the code or adjusting resource allocation to reduce the risk of errors.

By combining these two approaches, programmers could potentially reduce the risk of resource limitation errors in their code and optimize its performance. However, this approach would require significant development of both static and dynamic analysis tools and integration between them, making it a challenging but promising area for future research.

Figure 6. Research ideas for a topic are generated by ChatGPT.

4. Programming Learning with ChatGPT

The importance of computer programming in professional and academic fields is significant. Experienced programmers have demonstrated improvement in both professional and academic fields [52]. Programming skills are acquired through repeated practice. To assist programmers, deep-learning-based tools are introduced for code repair, completion, error detection, optimization, verification, and classification [1,2,19,20,53,54]. In recent years, LLMs based on the Transformer architecture (e.g., CodeBERT, Codex, and PyMT5) have achieved state-of-the-art results for various programming tasks [55]. ChatGPT is an LLM based on the Transformer architecture and has received significant attention because of its human-like conversational style. The application of ChatGPT is not limited to language-related tasks but also finds use in programming learning applications (e.g., code suggestion, optimization, completion, and error detection). However, the quality and suitability of these applications for programming learning remain unclear. Therefore, we evaluate and analyze the performance of ChatGPT in various programming learning tasks as follows.

4.1. Conceptual Understanding

A clear concept is a basic requirement for improved programming performance. ChatGPT can provide explanations and examples of various programming concepts (e.g., data structures, algorithms, languages, and programming language syntax) in a concise, simple, and understandable manner. For example, when we asked ChatGPT to “**provide a pseudocode for a selection sort and an explanation**”, it generated the pseudocode and explanation, as depicted in Figure 7. The results demonstrate that ChatGPT can generate easy-to-understand explanations and pseudocode that are useful for learners to understand algorithmic concepts.



Provide a pseudocode of selection sort and explanation



Sure, here's an example pseudocode for selection sort:

```

function selectionSort(array):
  for i from 0 to length(array) - 1 do
    // Find the minimum element in the unsorted portion of the array
    minIndex = i
    for j from i + 1 to length(array) do
      if array[j] < array[minIndex] then
        minIndex = j

    // Swap the minimum element with the first element in the unsorted portion
    if minIndex != i then
      temp = array[i]
      array[i] = array[minIndex]
      array[minIndex] = temp

  return array
  
```

Selection sort is a simple sorting algorithm that works by repeatedly finding the minimum element from the unsorted portion of an array and placing it at the beginning of the unsorted portion. The algorithm iterates over the entire array, selecting the smallest element from the unsorted portion each time and swapping it with the first element in the unsorted portion.

The outer loop of the selection sort algorithm iterates over each index of the array, from 0 to the length of the array minus 1. For each index, the inner loop finds the minimum element in the unsorted portion of the array by comparing each element in the unsorted portion with the current minimum. If a smaller element is found, its index is stored in 'minIndex'. Once the inner loop completes, the minimum element is swapped with the first element in the unsorted portion of the array, effectively expanding the sorted portion of the array by one element.

Selection sort has a time complexity of $O(n^2)$, which means that it can become quite slow for larger arrays. However, it is simple to understand and implement, and can be useful in certain situations where simplicity and clarity are more important than efficiency.

(a)

(b)

Figure 7. Pseudocode and explanation of selection sort algorithm. (a) Pseudocode of selection sort. (b) Explanation of selection sort.

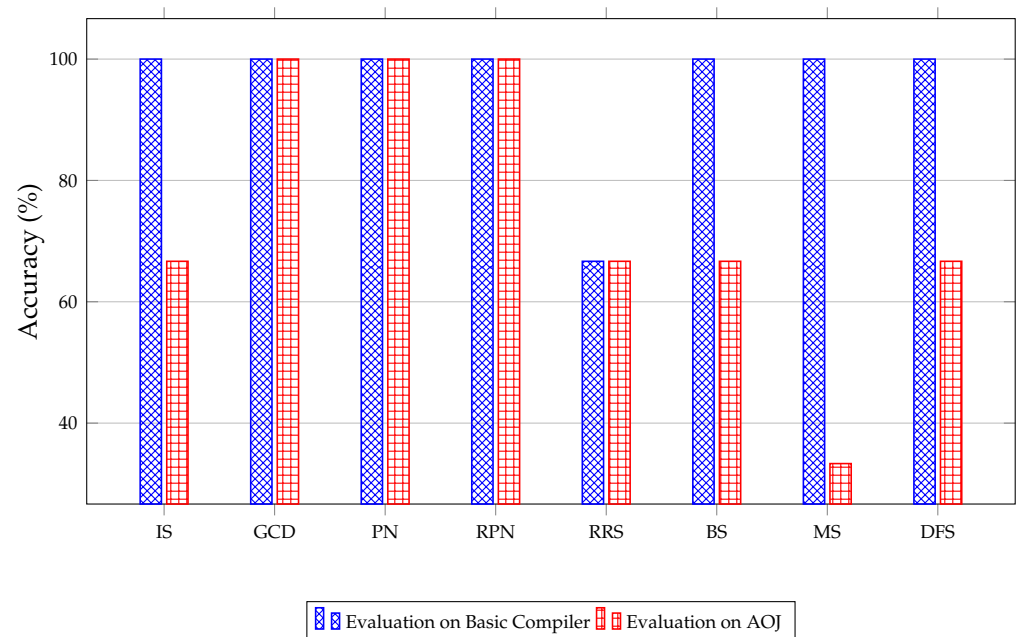
4.2. Solution Code Generation

Programmers can use ChatGPT to generate solution codes based on a problem description. Such solution codes can assist programmers in their programming learning phase. Therefore, we experimented to evaluate the performance of ChatGPT in generating code based on problem descriptions. In the experiment, we used the “**Algorithms and Data Structures**” problem description from the Aizu Online Judge (AOJ) system [56] and verified the correctness of the codes generated by ChatGPT on the same platform and in a basic compiler. We leveraged the problem descriptions of eight random problems and generated codes three times for each problem on the basis of the same description. We then used these strategies to validate our experimental results. Table 2 shows the correctness of the generated codes based on the problem descriptions using ChatGPT.

In addition, Figure 8 shows the comparative accuracy of the generated codes executed in the AOJ platform and a basic compiler. The following observations can be drawn: (i) the correctness rate of the generated code based on the basic compiler is approximately 95.83%; (ii) the correctness rate based on AOJ compilation is approximately 75%; (iii) when running a submitted code on the AOJ platform, various constraints (time, memory, etc.) and output formatting are taken into account, which may be a reason for the lower accuracy on the AOJ platform than the basic compiler; (iv) ChatGPT generates mostly correct code considering long problem descriptions, including constraints, algorithms, and input and output formatting.

Table 2. Evaluation results of generated codes based on Basic Compiler and AOJ system.

Case No.	Problem Name	Evaluation on Basic Compiler		Evaluation on AOJ	
		Success	Failure	Success	Failure
1	Insertion Sort (IS) (https://onlinejudge.u-aizu.ac.jp/courses/lesson/1/ALDS1/1/ALDS1_1_A accessed on 5 January 2023)	3	0	2	1 (WA)
2	GCD (https://onlinejudge.u-aizu.ac.jp/courses/lesson/1/ALDS1/1/ALDS1_1_B accessed on 5 January 2023)	3	0	3	0
3	Prime Numbers (PN) (https://onlinejudge.u-aizu.ac.jp/courses/lesson/1/ALDS1/1/ALDS1_1_C accessed on 13 January 2023)	3	0	3	0
4	Reverse Polish Notation (RPN) (https://onlinejudge.u-aizu.ac.jp/courses/lesson/1/ALDS1/1/ALDS1_3_A accessed on 13 January 2023)	3	0	3	0
5	Round-Robin Scheduler (RRS) (https://onlinejudge.u-aizu.ac.jp/courses/lesson/1/ALDS1/1/ALDS1_3_B accessed on 13 January 2023)	2	1	2	1 (RE)
6	Binary Search (BS) (https://onlinejudge.u-aizu.ac.jp/courses/lesson/1/ALDS1/1/ALDS1_4_B accessed on 05 February 2023)	3	0	2	1 (TLE)
7	Merge Sort (MS) (https://onlinejudge.u-aizu.ac.jp/courses/lesson/1/ALDS1/1/ALDS1_5_B accessed on 05 February 2023)	3	0	1	2 (TLE, WA)
8	Depth First Search (DFS) (https://onlinejudge.u-aizu.ac.jp/courses/lesson/1/ALDS1/1/ALDS1_11_B accessed on 10 February 2023)	3	0	2	1 (WA)

**Figure 8.** Comparison of the evaluation accuracy of the generated code for all problems executed with basic compiler and the AOJ platform.

4.3. Error Checking and Debugging in Code

Error checking and debugging code is a tedious and time-consuming task for learners, requiring them to check code line-by-line to identify errors and their locations. ChatGPT can identify errors in code and provide potential suggestions and code snippets. We experimented with ChatGPT to debug erroneous codes. To do this, we collected erroneous codes from the AOJ system. We asked the questions “Does this code have a bug? How can it be fixed?” to ChatGPT, and it responded with valuable suggestions, code snippets, and occasionally whole codes, as depicted in Figure 9. The suggestions proved to be interesting to the programmers and helpful in resolving the code errors.

Does this code have error? how to fix it?

```

1  #include<iostream>
2  #include<vector>
3  using namespace std;
4  #define MAX 2000000
5  #define INF 1000000
6  int H=0;
7  vector<int> A(MAX);
8  string com(1, 'a');
9  void heapIncreaseKey(int key){
10     int i = H;
11     A.at(i) = key;
12     while(i > 1 && A.at(i) > A.at(i / 2)){
13         swap(A.at(i), A.at(i / 2));
14         i /= 2;
15     }
16     return;
17 }
18 void insert(int key){
19     H++;
20     heapIncreaseKey(key);
21     return;
22 }
23 void maxHeapify(int i){
24     int l = 2*i;
25     int r = 2*i + 1;
26     int largest;
27     if (l <= H && A.at(l) > A.at(i)){
28         largest = l;
29     }else{
30         largest = i;
31     }
32     if (r <= H && A.at(r) > A.at(largest)){
33         largest = r;
34     }
35     if(A.at(largest) != A.at(i)){
36         swap(A.at(i), A.at(largest));
37         maxHeapify(largest);
38     }
39     return;
40 }
41 void buildMaxHeap(){
42     for(int i = H/2; i; i--){
43         maxHeapify(i);
44     }
45 }
46 }
47 void extract(){
48     cout << A.at(1) << endl;
49     A.at(1) = A.at(H+1) = -INF;
50     buildMaxHeap();
51     H--;
52     return;
53 }
54 int main(){
55     int n;
56     while(1){
57         cin >> com;
58         if(com.at(0) == 'i'){
59             cin >> n;
60             insert(n);
61         }
62         else if(com.length() == 3){
63             break;
64         }
65         else{
66             extract();
67         }
68     }
69     return 0;
70 }

```



The code has a few errors:

1. The function `extract()` is defined as returning `void`, but it should be returning an integer, which is the value of the extracted element.
2. The line `A.at(1) = A.at(H+1) = -INF;` in `extract()` is incorrect. It should be `A.at(1) = A.at(H); A.at(H) = -INF;`
3. The function `heapIncreaseKey()` is assuming that the heap starts at index 1, but the vector `A` is indexed from 0. So, the while loop condition should be `while(i > 0 && A.at(i) > A.at((i - 1) / 2))` instead of `while(i > 1 && A.at(i) > A.at(i / 2))`. Similarly, the line `if(A.at(largest) != A.at(i))` in `maxHeapify()` should be `if(largest != i)`.

Figure 9. Error checking and debugging in code using ChatGPT.

Of the participating students, 61.3% (35.5% first year, 16.1% second year, 6.5% third year, and 3.2% fourth year) were undergraduate students, 32.2% (16.1% first year and 16.1% second year) were Master's students, and 6.5% were second-year doctoral students. Figure 11 depicts the students' participation in this survey.

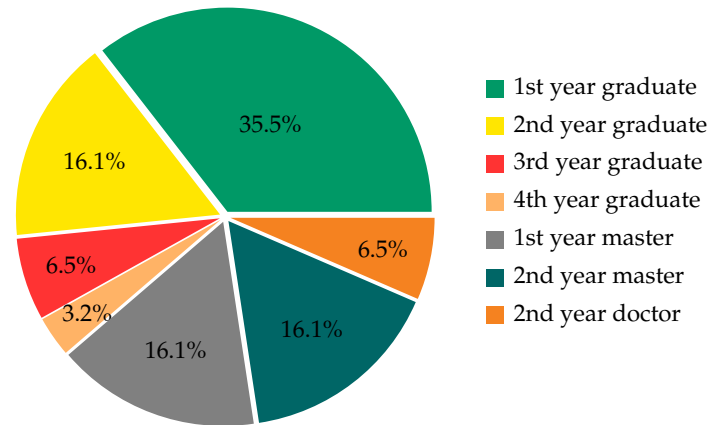


Figure 11. Student participation statistics in the survey.

We asked students to rate their programming experiences on a scale of 1 to 5, and received a score of 1/5 from about 29% of students, 2/5 from 25.8%, 3/5 from 25.8%, 4/5 from 9.7%, and 5/5 from 9.7%. We found that most students rated their programming experiences between 1/5 and 3/5. Figure 12 shows the programming skills of the participating students.

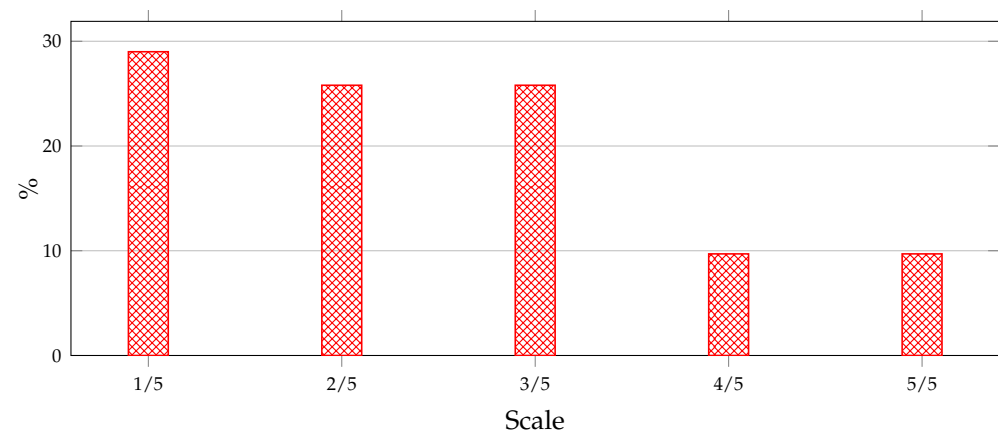


Figure 12. Programming experience (on a scale of 1 to 5) of the students who participated in the survey.

In response to the question "Have you taken help/support from ChatGPT for solving programming problems?", 78.8% of the students answered "Yes" and 21.2% answered "No". These results show that a large number of students have taken the support of ChatGPT for their programming learning. By asking the question "Do ChatGPT's suggestions help you in solving programming problems?", we tried to assess how ChatGPT suggestions help students. We found that about 86.7% of the students answered "Yes". Moreover, we asked students "What kind of problems have you solved with the help of ChatGPT?" and gave them two options, "simple problem" and "complex problem". The answers were interesting: about 59.3% of students used ChatGPT to solve their "complex" problems and about 40.7% of students used ChatGPT to solve "simple" problems. To evaluate the usefulness of ChatGPT for learning programming, 92.9% of the students answered "Yes". Finally, we asked students, "How satisfied are you with the use of ChatGPT? Please give a rating on a scale 1–5." and received a rating of 1/5 from 3.3% of students, 2/5 from 10%, 3/5 from 30%, 4/4 from 36.6%, and 5/5

from 20%. It can be seen that most students were satisfied with the programming support provided by ChatGPT.

On the other hand, a similar type of survey was conducted with teachers to evaluate how ChatGPT supports programming teaching and research. To the question “*Have you taken help/support from ChatGPT for teaching programming?*”, about 60% of the teachers answered “Yes”. For questions about programming teaching and research, most teachers were satisfied with the suggestions provided by ChatGPT.

6. Threats and Strategies

As an AI LLM, ChatGPT can play an important role in education and research. The capabilities of this powerful tool are not limited to these fields but also apply to many others. However, despite the many advantages of using ChatGPT, there are also challenges in using it for education and research, especially for technical education such as computer programming. Because ChatGPT is capable of generating texts that are nearly indistinguishable from human-generated texts in high-level cognitive tasks, this capability of ChatGPT raises concerns about its potential use in education and research. In this section, we present the challenges and possible strategies for using ChatGPT.

Integrity of assignments and online exams: Online exams have become a common phenomenon in higher education. As ChatGPT can generate human-like text for academic topics, educators and institutions need to be aware of the possibility of cheating in online exams using ChatGPT. In short, ChatGPT threatens the fairness and validity of online exams and assignments. To address these raised issues, there are some strategies that educators and institutions can take. Students can be given clear instructions on assignments and online exams on how to structure their assignments and answer their questions online [57]. Students can send their assignments to teachers for review before final submission. An advanced plagiarism-detection tool can be used to detect AI-generated texts. Furthermore, advanced exam supervision/proctoring techniques could be effective for online exams [14]. In this context, further research is required to fully understand the impact of AI LLMs such as ChatGPT and strategies for combating the misuse of ChatGPT.

Blind reliance on generative AI tools: A heavy reliance on generative AI tools such as ChatGPT can negatively affect education and research. This is because the ease of obtaining answers, problem-solving strategies, and scientific text generation can limit critical thinking and problem-solving skills. Recently, ChatGPT has authored and credited published papers and preprints [58]. It also raises questions about writing essays and research articles. The CEO of OpenAI warns against blind reliance on ChatGPT, saying:

“ChatGPT is incredibly limited but good enough at some things to create a misleading impression of greatness. It’s a mistake to be relying on it for anything important but a preview of progress. We have lots of work to do on robustness and truthfulness.”

— Sam Altman, CEO of OpenAI

To address this issue, it is important for students, educators, and researchers to be aware of the limitations of LLMs, and these tools can only be used as supportive tools to enhance research and learning [59]. Further research is required to design academic curricula, question-and-answer patterns, assignments, and exams to address the challenges raised.

Difficulty in evaluating the ChatGPT-generated answers and texts: As an AI LLM, ChatGPT uses complex algorithms and statistical models to generate answers and text on the basis of patterns learned from large amounts of text data. The answers and texts generated by ChatGPT are becoming indistinguishable from human-generated answers and texts. This poses a challenge to educators and researchers [57,60–62]. Existing plagiarism-detection tools are finding it increasingly difficult to distinguish between AI- and human-generated texts. As a result, restrictions have been placed on the use of ChatGPT in educational institutions [63]. Cotton et al. [57] presented several strategies for recognizing texts from LLMs such as ChatGPT, including language inconsistencies, a lack of proper citations, factual errors, ambiguity, and poor context awareness. Further research on the

development of new technologies (e.g., AI-based plagiarism detectors) is needed to ensure the integrity of education and research.

Ethical implications and potential biases: The ethical implications and biases of using ChatGPT in education and research should be carefully considered. Typically, LLMs rely heavily on training data, and when the data contain biases or anomalies, it could lead to unfair results. For example, if the training data are biased toward certain people or cultures, the model may produce unfair or discriminatory output. Therefore, it is imperative to ensure that the training data are diverse and well-balanced. ChatGPT and other AI language models can be used to generate fake news, hate speech, and other harmful content. This can lead to social unrest, reputation damage, and even physical harm. Furthermore, the internal mechanisms and processes are not sufficiently open and transparent to users about how they work. It is also important to ensure that the decision-making processes of these models are transparent to users. Because ChatGPT generates responses without human intervention, it can be difficult to hold anyone accountable for the responses generated. This may make it difficult to address any ethical concerns or biases. ChatGPT and other generative models involve the collection and processing of personal data, which raises concerns about privacy and data security. Appropriate measures should be taken to protect unauthorized access to individual data.

Critical thinking and problem-solving skill: ChatGPT can generate nearly accurate answers to technical questions from a wide range of topics and correct or partially correct programming code based on problem descriptions, algorithm and problem names, etc. Simply acquiring answers and code from ChatGPT can be a barrier to improving learners' critical thinking and problem-solving skills. To the best of our knowledge, there are no such tools that can recognize code generated by AI models, and thus, the solution codes generated by AI models can be used for academic coding exams and competitions. This poses a challenge to educators on how to deal with this new situation.

However, there are some strategies for determining whether responses and programming codes are generated by ChatGPT. *Look for telltale signs:* ChatGPT responses typically have certain characteristics, such as a lack of personalization or a rather generic tone. Furthermore, the programming code typically contains programming syntax and formatting. *Check for coherence:* ChatGPT responses may not have a consistent or logical flow, especially when it is generating answers to complex questions. If the answers appear disjointed or nonsensical, this may indicate that they were generated by ChatGPT or another AI model. *Compare responses:* We can compare responses generated by ChatGPT with responses generated by other language models or humans. If the answer is identical to answers generated by ChatGPT, it may be a sign that the answer was not generated by a human. *Use of plagiarism-detection tools:* we may also use plagiarism-detection tools to determine whether an answer contains programming code copied from somewhere else. This can help detect cases of fraud. In addition, if there is possible cheating with programming codes, we can ask follow-up questions to determine the depth of the learner's understanding in answering the question.

7. Limitations

It is worth noting that our experiments were conducted on ChatGPT, which is currently in an active development phase. During the experiments, we obtained significant results in code generation, error checking and debugging, and optimization of the solution code. The results may vary for the following reasons: (i) the release of a new version of ChatGPT may lead to different results; (ii) asking different questions than those presented in this study; (iii) results may vary for different problem descriptions; (iv) code-optimization results may vary for different solution codes.

Apart from that, the experimental results show that the accuracy of the generated code is 95.83% on the basic compiler and 75% on AOJ. The average accuracy of the code generated by ChatGPT is 85.42%. Although ChatGPT generates about 85.42% correct code, it still has the limitations of generating code based on the description. In some cases,

adjustments to the generated code may be necessary to achieve error-free compilation and acceptance.

Furthermore, in the survey for teachers, we asked the question, “How satisfied are you with the use of ChatGPT? Please give a rating on a scale of 1–5”. About 50% of teachers gave a rating of 3/5. This result also indicates that ChatGPT has not yet gained enough trust in programming education. Some teachers shared their real-life experiences, such as “ChatGPT is still not perfect to answer some exams of my Java class”. In addition, we asked ChatGPT to solve an elementary school quiz that involved counting the number of occurrences of the digit “8” in a given number “348789623489109823647864351672”, and it failed every time to count the exact number of occurrences of the digit “8”, which can be easily counted by elementary students. It also shows the clear limitation of ChatGPT. There is a risk of blindly trusting ChatGPT.

8. Conclusions

ChatGPT and other AI LLMs have the potential and can be supporting tools for educational and research work. ChatGPT is a revolutionary LLM that can maintain human-like conversations and generate human-like text for any natural language query that is nearly indistinguishable. The model can be used to answer questions, write essays, solve problems, explain complex topics, provide virtual tutoring, practice languages, learn programming, teach, and support research. Furthermore, the ChatGPT model can be used to solve technical (e.g., engineering and computer programming) and non-technical (e.g., language and literature) problems. Our surveys and experimental results show that ChatGPT is useful not only for programming education but also for education and research. However, although ChatGPT is a powerful tool that can generate impressive responses on a variety of topics, it still has certain limitations, such as a lack of common sense, potential bias, difficulty with complex reasoning, and inability to process visual information. It is important to keep in mind the limitations of ChatGPT when using it, and it should not be relied upon blindly. In addition, the ethical implications (e.g., bias and discrimination, privacy and security, misuse of technology, accountability, transparency, and social impact) of ChatGPT are complex and multifaceted and should be carefully considered.

Despite the various difficulties and challenges, we believe that the risks discussed can be effectively managed and must be addressed to provide reliable and equitable access to LLMs for educational and research purposes.

Author Contributions: Conceptualization, M.M.R. and Y.W.; Data curation, M.M.R.; Formal analysis, M.M.R.; Funding acquisition, Y.W.; Methodology, M.M.R.; Project administration, Y.W.; Resources, M.M.R.; Supervision, Y.W.; Validation, M.M.R.; Visualization, M.M.R.; Writing—original draft, M.M.R.; Writing—review and editing, Y.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Japan Society for the Promotion of Science (JSPS) KAKENHI (<https://kaken.nii.ac.jp/en/grant/KAKENHI-PROJECT-23H03508/> accessed on 15 March 2023). Grant Number: 23H03508.

Institutional Review Board Statement: Not applicable

Informed Consent Statement: Not applicable

Data Availability Statement: The experimental dataset, including all problem descriptions and erroneous source code, was collected from the Aizu Online Judge System. The web links are: <http://developers.u-aizu.ac.jp/index> (accessed on 15 March 2023) and <https://onlinejudge.u-aizu.ac.jp/> (accessed on 15 March 2023).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Rahman, M.M.; Watanobe, Y.; Nakamura, K. A neural network based intelligent support model for program code completion. *Sci. Program.* **2020**, *2020*, 7426461. [\[CrossRef\]](#)
2. Rahman, M.M.; Watanobe, Y.; Nakamura, K. A bidirectional LSTM language model for code evaluation and repair. *Symmetry* **2021**, *13*, 247. [\[CrossRef\]](#)
3. Rahman, M.M. Data Analysis and Code Assessment Using Machine Learning Techniques for Programming Activities. Ph.D. Thesis, The University of Aizu, Aizuwakamatsu, Japan, 2022.
4. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*. Available online: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf (accessed on 15 March 2023).
5. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 1877–1901.
6. Floridi, L.; Chiriatti, M. GPT-3: Its nature, scope, limits, and consequences. *Minds Mach.* **2020**, *30*, 681–694. [\[CrossRef\]](#)
7. OpenAI-Team. ChatGPT: Optimizing Language Models for Dialogue. 2022. Available online: <https://openai.com/blog/chatgpt/> (accessed on 11 March 2023).
8. Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.L.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. Training language models to follow instructions with human feedback. *arXiv* **2022**, arXiv:2203.02155.
9. Kasneci, E.; Seßler, K.; Küchemann, S.; Bannert, M.; Dementieva, D.; Fischer, F.; Gasser, U.; Groh, G.; Günnemann, S.; Hüllermeier, E.; et al. ChatGPT for good? On opportunities and challenges of large language models for education. *Learn. Individ. Differ.* **2023**, *103*, 102274. [\[CrossRef\]](#)
10. Qadir, J. Engineering Education in the Era of ChatGPT: Promise and Pitfalls of Generative AI for Education. *TechRxiv Prepr.* **2022**. [\[CrossRef\]](#)
11. Thunstrom, A.O. We asked GPT-3 to write an academic paper about itself: Then we tried to get it published. *Scientific American*, 30 June 2022.
12. Stokel-Walker, C. AI bot ChatGPT writes smart essays-should academics worry? *Nature*, 9 December 2022.
13. Welsh, M. The End of Programming. *Commun. ACM* **2022**, *66*, 34–35. [\[CrossRef\]](#)
14. Susnjak, T. ChatGPT: The End of Online Exam Integrity? *arXiv* **2022**, arXiv:2212.09292.
15. Parslow, G.R. Commentary: How the internet is changing the way we think, read and remember. *Biochem. Mol. Biol. Educ.* **2011**, *39*, 228. [\[CrossRef\]](#) [\[PubMed\]](#)
16. Pappano, L. The Year of the MOOC. *New York Times*, 4 November 2012.
17. Wollny, S.; Schneider, J.; Di Mitri, D.; Weidlich, J.; Rittberger, M.; Drachslar, H. Are we there yet?—A systematic literature review on chatbots in education. *Front. Artif. Intell.* **2021**, *4*, 654924. [\[CrossRef\]](#) [\[PubMed\]](#)
18. Rahman, M.M.; Watanobe, Y.; Rage, U.K.; Nakamura, K. A novel rule-based online judge recommender system to promote computer programming education. In Proceedings of the Advances and Trends in Artificial Intelligence. From Theory to Practice: 34th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2021, Kuala Lumpur, Malaysia, 26–29 July 2021; pp. 15–27.
19. Rahman, M.M.; Watanobe, Y.; Nakamura, K. Source code assessment and classification based on estimated error probability using attentive LSTM language model and its application in programming education. *Appl. Sci.* **2020**, *10*, 2973. [\[CrossRef\]](#)
20. Rahman, M.M.; Watanobe, Y.; Kiran, R.U.; Kabir, R. A stacked bidirectional lstm model for classifying source codes built in mpls. In Proceedings of the Machine Learning and Principles and Practice of Knowledge Discovery in Databases: International Workshops of ECML PKDD 2021, Virtual Event, 13–17 September 2021; pp. 75–89.
21. Litman, D. Natural language processing for enhancing teaching and learning. In Proceedings of the AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016; Volume 30.
22. Alshater, M.M. Exploring the role of artificial intelligence in enhancing academic performance: A case study of ChatGPT. *SSRN* **2022**. [\[CrossRef\]](#)
23. Dowling, M.; Lucey, B. ChatGPT for (finance) research: The Bananarama conjecture. *Financ. Res. Lett.* **2023**, 103662. [\[CrossRef\]](#)
24. Rudolph, J.; Tan, S.; Tan, S. ChatGPT: Bullshit spewer or the end of traditional assessments in higher education? *J. Appl. Learn. Teach.* **2023**, *6*. [\[CrossRef\]](#)
25. Frieder, S.; Pinchetti, L.; Griffiths, R.R.; Salvatori, T.; Lukasiewicz, T.; Petersen, P.C.; Chevalier, A.; Berner, J. Mathematical capabilities of chatgpt. *arXiv* **2023**, arXiv:2301.13867.
26. Kung, T.H.; Cheatham, M.; Medenilla, A.; Sillos, C.; De Leon, L.; Elepaño, C.; Madriaga, M.; Aggabao, R.; Diaz-Candido, G.; Maningo, J.; et al. Performance of ChatGPT on USMLE: Potential for AI-assisted medical education using large language models. *PLoS Digit. Health* **2023**, *2*, e0000198. [\[CrossRef\]](#)
27. Gilson, A.; Safranek, C.; Huang, T.; Socrates, V.; Chi, L.; Taylor, R.A.; Chartash, D. How Well Does ChatGPT Do When Taking the Medical Licensing Exams? The Implications of Large Language Models for Medical Education and Knowledge Assessment. *medRxiv* **2022**.
28. Aydın, Ö.; Karaarslan, E. OpenAI ChatGPT generated literature review: Digital twin in healthcare. *SSRN* **2022**. [\[CrossRef\]](#)

29. Watanobe, Y.; Rahman, M.M.; Amin, M.F.I.; Kabir, R. Identifying algorithm in program code based on structural features using CNN classification model. *Appl. Intell.* **2022**, 1–27. [\[CrossRef\]](#)
30. Li, Y.; Choi, D.; Chung, J.; Kushman, N.; Schrittwieser, J.; Leblond, R.; Eccles, T.; Keeling, J.; Gimeno, F.; Dal Lago, A.; et al. Competition-level code generation with alphacode. *Science* **2022**, 378, 1092–1097. [\[CrossRef\]](#) [\[PubMed\]](#)
31. Castelvechi, D. Are ChatGPT and AlphaCode going to replace programmers? *Nature*, 8 December 2022.
32. Sobania, D.; Briesch, M.; Hanna, C.; Petke, J. An analysis of the automatic bug fixing performance of chatgpt. *arXiv* **2023**, arXiv:2301.08653.
33. Jalil, S.; Rafi, S.; LaToza, T.D.; Moran, K.; Lam, W. ChatGPT and Software Testing Education: Promises & Perils. *arXiv* **2023**, arXiv:2302.03287.
34. Malinka, K.; Perešini, M.; Firc, A.; Hujňák, O.; Januš, F. On the educational impact of ChatGPT: Is Artificial Intelligence ready to obtain a university degree? *arXiv* **2023**, arXiv:2303.11146.
35. Kashefi, A.; Mukerji, T. Chatgpt for programming numerical methods. *arXiv* **2023**, arXiv:2303.12093.
36. White, J.; Hays, S.; Fu, Q.; Spencer-Smith, J.; Schmidt, D.C. Chatgpt prompt patterns for improving code quality, refactoring, requirements elicitation, and software design. *arXiv* **2023**, arXiv:2303.07839.
37. Lee, H. The rise of ChatGPT: Exploring its potential in medical education. *Anat. Sci. Educ.* **2023**. [\[CrossRef\]](#)
38. Dai, H.; Liu, Z.; Liao, W.; Huang, X.; Wu, Z.; Zhao, L.; Liu, W.; Liu, N.; Li, S.; Zhu, D.; et al. Chataug: Leveraging chatgpt for text data augmentation. *arXiv* **2023**, arXiv:2302.13007.
39. Wang, F.Y.; Yang, J.; Wang, X.; Li, J.; Han, Q.L. Chat with chatgpt on industry 5.0: Learning and decision-making for intelligent industries. *IEEE/CAA J. Autom. Sin.* **2023**, 10, 831–834. [\[CrossRef\]](#)
40. Wollowski, M. Using ChatGPT to produce code for a typical college-level assignment. *AI Mag.* **2023**, 44, 129–130. [\[CrossRef\]](#)
41. Du, H.; Teng, S.; Chen, H.; Ma, J.; Wang, X.; Gou, C.; Li, B.; Ma, S.; Miao, Q.; Na, X.; et al. Chat with chatgpt on intelligent vehicles: An iee tiv perspective. *IEEE Trans. Intell. Veh.* **2023**, 8, 2020–2026. [\[CrossRef\]](#)
42. Shen, Y.; Song, K.; Tan, X.; Li, D.; Lu, W.; Zhuang, Y. Hugginggpt: Solving ai tasks with chatgpt and its friends in huggingface. *arXiv* **2023**, arXiv:2303.17580.
43. Avila-Chauvet, L.; Mejía, D.; Acosta Quiroz, C.O. Chatgpt as a Support Tool for Online Behavioral Task Programming. *SSRN* **2023**. Available online: <https://ssrn.com/abstract=4329020> (accessed on 15 March 2023).
44. Antaki, F.; Touma, S.; Milad, D.; El-Khoury, J.; Duval, R. Evaluating the performance of chatgpt in ophthalmology: An analysis of its successes and shortcomings. *medRxiv* **2023**. [\[CrossRef\]](#)
45. Rao, A.S.; Kim, J.; Kamineni, M.; Pang, M.; Lie, W.; Succi, M. Evaluating ChatGPT as an adjunct for radiologic decision-making. *medRxiv* **2023**. [\[CrossRef\]](#)
46. Wenzlaff, K.; Spaeth, S. Smarter than Humans? Validating how OpenAI’s ChatGPT model explains Crowdfunding, Alternative Finance and Community Finance. *SSRN* **2022**. [\[CrossRef\]](#)
47. Zaremba, A.; Demir, E. ChatGPT: Unlocking the Future of NLP in Finance. *SSRN* **2023**. [\[CrossRef\]](#)
48. Choi, J.H.; Hickman, K.E.; Monahan, A.; Schwarcz, D. Chatgpt goes to law school. *SSRN* **2023**. [\[CrossRef\]](#)
49. Jiao, W.; Wang, W.; Huang, J.t.; Wang, X.; Tu, Z. Is ChatGPT a good translator? A preliminary study. *arXiv* **2023**, arXiv:2301.08745.
50. Beth, M. AI and the Future of Undergraduate Writing. 2022. Available online: <https://www.chronicle.com/article/ai-and-the-future-of-undergraduate-writing> (accessed on 12 March 2023).
51. Sharples, M. Automated essay writing: An AIED opinion. *Int. J. Artif. Intell. Educ.* **2022**, 32, 1119–1126. [\[CrossRef\]](#)
52. Rahman, M.M.; Watanobe, Y.; Kiran, R.U.; Thang, T.C.; Paik, I. Impact of practical skills on academic performance: A data-driven analysis. *IEEE Access* **2021**, 9, 139975–139993. [\[CrossRef\]](#)
53. Zhang, Q.; Fang, C.; Ma, Y.; Sun, W.; Chen, Z. A Survey of Learning-based Automated Program Repair. *arXiv* **2023**, arXiv:2301.03270.
54. Rahman, M.M.; Watanobe, Y.; Nakamura, K. Evaluation of source codes using bidirectional lstm neural network. In Proceedings of the 2020 3rd IEEE International Conference on Knowledge Innovation and Invention (ICKII), Kaohsiung, Taiwan, 21–23 August 2020; pp. 140–143.
55. Sobania, D.; Briesch, M.; Rothlauf, F. Choose your programming copilot: A comparison of the program synthesis performance of github copilot and genetic programming. In Proceedings of the Genetic and Evolutionary Computation Conference, New York, NY, USA, 9–13 July 2022; pp. 1019–1027.
56. Watanobe, Y.; Rahman, M.M.; Matsumoto, T.; Rage, U.K.; Ravikumar, P. Online judge system: Requirements, architecture, and experiences. *Int. J. Softw. Eng. Knowl. Eng.* **2022**, 32, 917–946. [\[CrossRef\]](#)
57. Cotton, D.R.; Cotton, P.A.; Shipway, J.R. Chatting and Cheating. Ensuring academic integrity in the era of ChatGPT. *Innov. Educ. Teach. Int.* **2023**, 1–12. [\[CrossRef\]](#)
58. Stokel-Walker, C. ChatGPT listed as author on research papers: Many scientists disapprove. *Nature* **2023**, 613, 620–621. [\[CrossRef\]](#)
59. Pavlik, J.V. Collaborating with ChatGPT: Considering the Implications of Generative Artificial Intelligence for Journalism and Media Education. *J. Mass Commun. Educ.* **2023**, 78, 84–93. [\[CrossRef\]](#)
60. Elkins, K.; Chun, J. Can GPT-3 pass a Writer’s turing test? *J. Cult. Anal.* **2020**, 5, 1–16. [\[CrossRef\]](#)
61. Gao, C.A.; Howard, F.M.; Markov, N.S.; Dyer, E.C.; Ramesh, S.; Luo, Y.; Pearson, A.T. Comparing scientific abstracts generated by ChatGPT to original abstracts using an artificial intelligence output detector, plagiarism detector, and blinded human reviewers. *bioRxiv* **2022**. [\[CrossRef\]](#)

62. Dehouche, N. Plagiarism in the age of massive Generative Pre-trained Transformers (GPT-3). *Ethics Sci. Environ. Politics* **2021**, *21*, 17–23. [[CrossRef](#)]
63. Kalhan, R. ChatGPT Banned from New York City Public Schools' Devices and Networks. 2023. Available online: <https://www.nbcnews.com/tech/tech-news/new-york-city-public-schools-ban-chatgpt-devices-networks-rcna64446> (accessed on 12 March 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.