VAWKUM Transactions on Computer Sciences
http://vfast.org/journals/index.php/VTCS@ 2025, ISSN(e): 2308-8168, ISSN(p): 2411-6335
Volume 13, Number 1, January-June 2025 *pp:201-216*

AI vs. Human Programmers: Complexity and Performance in Code Generation

Samina Azeem [1*], **Muhammad Shumail Naveed** [2], **Muhammad Sajid** [2], **Imran Ali** [2]

[1]Department of Computer Science, Sardar Bahadur Khan Women University, Pakistan;
[2]Department of Computer Science & Information Technology, University of Balochistan, Quetta, Pakistan

Keywords: *Artificial Intelligence, ChatGPT, Programming, Halstead Complexity, Cyclomatic Complexity.*

Journal Info:
Submitted:
January 14, 2025
Accepted:
April 30, 2025
Published:
May 10, 2025

Abstract

Large language models, like ChatGPT, have shown the ability to do a variety of tasks in different fields, and this has increased efficiency greatly. However, their increasing use is causing concern about the potential job displacement, particularly in the technical fields. While there have been many studies on the performance of large language models in technical fields, there is a notable absence in assessing their performances in programming. This study fills this gap by comparing ChatGPT (GPT-4) and human experts in the coding discipline to determine if ChatGPT has advanced to a point where it can replace human programmers. To accomplish this goal, this study has produced 300 Python programs with ChatGPT (GPT-4) and compared them with functionally equivalent programs written by three experienced human programmers. The evaluation included both quantitative and qualitative evaluations using measures such as Halstead Complexity, Cyclomatic Complexity, and expert judgment by two human evaluators. The results showed statistically significant differences between the ChatGPT-generated and human-written code. Programs that were generated by ChatGPT were shown to be verbose, complex, and resource demanding, which is reflected in higher program volume, difficulty, and cyclomatic complexity scores. In qualitative terms, ChatGPT's code was easier to read, but lagged behind in some key areas, such as the quality of documentation, structuring of functions, and compliance with coding standards. On the other hand, human-written programs performed well in terms of maintainability, error handling, and dealing with edge cases. Although ChatGPT was found to be incredibly efficient at creating working code, the output needed a lot of review and refinement to be considered standard. The study concluded while ChatGPT is a useful tool for generating code, it has not yet reached the level needed to replace human expertise in programming.

***Correspondence author email address:** saminajehenzeib@gmail.com
DOI: 10.21015/vtcs.v13i1.2043

1 Introduction

Information technology is now an indispensable component of the worldwide society and it has penetrated almost every facet of modern life [1]. At the center of this technological revolution is the computer - a universally applicable, automated machine that is controlled by programs. Software, which is a collection of these programs, acts as the tool for computers to function and provide their amazing capabilities [2]. These programs are the backbone to innovation, being the critical interface between human intention and machine capability. Consequently, one cannot overstate the role that programming plays in the advancement of technology.

In the context of computer science, the term programming is no longer used for the main operations of a computer but for the process of writing instructions or algorithms based on the needs of a specific user. These instructions when translated into a language understandable by the machine allow for desired outcomes [3]. As a cornerstone of computational systems, programming is the basis of almost all technological breakthroughs, the critical framework on which advancements are built.

In computer science, programming is one of the foundations for fundamental concepts, and an essential skill for students to acquire in order to obtain a computer science degree [4]. The need for programmers is expected to increase by 8% over the period 2012 to 2022 with software developer jobs expected to increase highly by 2026 [2]. Programming is a crucial part of various fields such as software development, technology, engineering and even simple tools that are used daily, such as smartphones. It fosters critical thinking, innovativeness, and problem-solving ability, which are significant talents in the current techno world [5]. Further, programming finds applications in other sectors as well, and is remotely applied to a wide range of purposes rather often. Its transformational influence continues to expand with an increasing number of individuals becoming coders. Programming has not only been the driver behind the innovation of technology but there is an assurance of the same in the future. Nonetheless, the difficulties of learning how to program are enormous; it is labor intensive and is very time-consuming and requires very high ability to think abstractly in order to have a profound knowledge. To address these problems, the traditional way of teaching is evolving and embracing new techniques of enhancing the learning activities. It is significant to mix the old methods of instruction with the new ones in order to make the programming education more pleasant and accessible. As a result, there is a pressing necessity to find the instructional strategies that will effectively integrate the classical and the modern teaching methods and encourage the learners to achieve improved results.

Nowadays, programming has become one of the pillars of technological innovation, yet there are new directions in the development of the artificial intelligence (AI) that can potentially enhance the programming process. AI has been a game-changer as it has accelerated the future in most industries due to its speedy development in technologies that underpin it. The main branches of AI are Machine Learning and Deep learning and such processes as Natural language Generator (NLG), voice and speech recognition, robotics, and biometrics [6].

Large Language Models (LLMs) are an innovative AI trend that started in the area of deep learning, which is a machine learning subdiscipline. With their widespread abilities and uses, these models are reshaping the industries and day-to-day life. It is worth noting that using models like ChatGPT, Codex, Gemini, and Google T5, companies are contributing to the development in many fields, which shows how they play a central part in the continued development of AI [7].

The Transformer architecture is a complex neural network architecture built upon which large language models are built on and has been known to be highly scalable and heavily parameterized. The self-attention mechanism is central to the Transformer, and one of the innovations that allow these models to capture and comprehend interpretations between elements in an input sequence effectively [8].

Large Language Models (LLMs) and AI in general have become one of the major catalysts of technological and scientific progress, with applications in health, finance, agriculture, and education among others being notable

challenges to the medium-term outlook of these industries [9]. In the medical field, AI benefits by improving the diagnosis of patients, automating processes in the healthcare institution, and increasing the accuracy of the surgery procedure [10]. The data analysis process with the help of AI is also much faster, speeding up the creation and the launch of new medications into the marketplace in general [11]. In farming, AI robotics help in crop regulation, forecasting production, and so on, making farming more efficient and aiding in producing more food products [12]. In the financial sector, AI is useful in predicting, enterprise valuation, and exchange rate forecasting [13]. Artificial intelligence based intelligent tutors are becoming a disruptive technology in the education field by providing personalized, efficient learning experiences based on individual requirements [14].

Technologies are always merged with other new emerging technologies to form new paradigms. To give an example, AI can be used together with such disciplines as blockchain [15–18] to operate on data in a safe way, machine learning [19], and deep learning [20–22] to detect insightful trends and federated learning [23, 24] to train distributed models without violating privacy. Such associations are revolutionising the whole industry as more automated, secure and efficient technologies are being made available.

ChatGPT and other large language models are simplifying the programming process by helping the developer with their code development, debugging, and optimization. Bucaioni et al.[25] grew to realize that ChatGPT is effective when solving simple and intermediate-level programs in programming languages such as C++ and Java, although it is limited, as noted by the author, to its core capabilities, which may not be extensively developed to address complex programming issues. Jain et al. [26] have emphasized that LLMs integrate well with human programmers, whereas Koubaa et al. [27] have proven that human programmers continue to have a significant advantage over ChatGPT on several aspects of problems concerning programming, as mentioned by them. Programming methodologies are now extensively powered by ChatGPT, and could be applied to programming too [28]. The rapid progress of the AI sphere is not only making human life more powerful but also increases the question of whether the human labor can also be replaced by the AI-driven technologies. These developments increase the threats of job replacement. It is generally said that AI applications may cause the replacement of human labors possibly [29]. AI technologies like chatbot: ChatGPT will be of immense importance in reshaping the labor market and present white-collar and creative employees with a risk of losing their jobs due to their robotisation and automation in tasks traditionally performed by humans alone [30]. Even such jobs that were deemed as safe as software development, now have the chance of being replaced by AI. It has become increasingly held that ChatGPT may turn out to be able to substitute humans in different technical fields sooner or later [31].

Since their inception and the day when they were first introduced to the public, many studies have examined whether or not large language models, including ChatGPT, have reached the point where it can actually replace human individuals in a specific field. As far as we know, no thorough research was carried out to determine whether ChatGPT and other AI systems can really help to substitute humans in the programming industry.

Developing a potential answer to the question in the context of ChatGPT and analogous AIs is whether they will be able to replace some of the functions of the programming world would be an interesting topic to address. It may allow seeing what AI does well in the process of writing codes and in what areas human skills have not yet been replaced. It is in that light that companies and tech giants might be in a better position to make decisions more mindful regarding how to introduce AI into their operations with an objective of achieving efficiency and quality. It might also inform trainers and practitioners in reformulating training curricular to concentrate on talents that complement AI. At a larger scale, such a study could assist in designing ethical principles and job-related policies regarding AI in software development. Finally, by simply understanding how AI works best, organizations would be able to automate repetitive work and leave human developers to devote more creativity and challenge to their work.

This paper is expected to fill this research gap by contrasting the programming skills of human programmers

with those of ChatGPT. This type of investigation is of paramount importance, both as it assists in solving the problem of human fears of job substitution by AI and offers important data on the strength and weaknesses of ChatGPT in the area of programming.

2 Related work

Meaningful studies have already been done in the comparison between the performance of AI tools, such as ChatGPT, with human professionals. But the majority of such studies are aimed at the interaction of AI and human specialists who collaborate instead of comparing these two factors to each other. Notably, there is a lack of substantial research comparing ChatGPT and human experts specifically in the domain of programming. The literature reviewed in this section examines studies that have analyzed the capabilities of ChatGPT across other prominent disciplines, providing context for understanding their potential and limitations.

Steiss et al. [32], conducted a study to evaluate ChatGPT's ability to provide formative feedback by comparing the quality of feedback generated by humans and AI on secondary student essays. The study analyzed 200 instances of human-generated feedback and 200 instances of AI-generated feedback for the same essays. The findings described that the quality of feedback varied between AI and human evaluators depending on the quality of the essay.Human raters have always demonstrated superiority over AI in terms of providing quality feedback in all aspects except where criteria-based assessment is concerned. The paper indicated that generative AI might be applicable in some scenarios, especially when it comes to receiving feedback on the initial draft or in areas where a trained teacher is unavailable. The general findings were that human evaluators with proper training provided superior feedback in comparison with ChatGPT.

Duong et al. [33], compared the ChatGPT with human respondents on the genetic queries. The authors evaluated ChatGPT in answering 85 multiple-choice questions that pertained to human genetics and compared its outputs to 13,642 human-generated responses. It was found that ChatGPT had an accuracy of 68.2%, marginally above the accuracy of human respondents of 66.6%. He found that ChatGPT and human subjects did more well on memorization tasks as opposed to critical thinking. Although ChatGPT showed excellent results, the research identified serious constraints that still obstruct its applicability to clinical settings or any other high-stake environments.

The study by Breithaupt et al. [34] explored the retelling of stories in ChatGPT through three retelling steps and compared the performance of the AI to that of human storytellers. ChatGPT was also discovered to create good summaries of the original narrative text during the first retelling with slight modifications in the subsequent retelling. By contrast, human retellers added 55-60% new words and concepts (synsets) every time they retold the same passage, and the original text became progressively shorter, but with extremely high originality rates. ChatGPT and human retellings were also consistent in their ratings of emotions across iterations and this was a challenge to human retellers because of the originality in their retelling. ChatGPT could remember more nouns, adjectives and prepositions, which are learned later in life, with linguistics whereas humans used more verbs, adverbs, and negations, which are learned earlier in life. The research findings deduced that though retellings by ChatGPT are based on probabilistic language models, retellings by human beings are characterized with persistent novelty that is deeply entrenched in emotional content, and this fact makes the difference between human and AI-generated storytelling.

Wang et al. [35], examined the amount of domain-specific knowledge in pathology that ChatGPT exhibits on two underlying large language models, GPT-3.5 and the updated GPT-4. The researchers used 15 pathologists that created pathology-specific questions similar to those that were used in licensing (board) exams. GPT-3.5, GPT-4, and staff pathologists who recently passed their Canadian licensing examination in pathology gave responses to the 15 questions. The participants were required to rate the responses using a 5-point Likert scale, and how likely

a particular response was created by ChatGPT. The findings showed that GPT-4 was better than both GPT-3.5 and the staff pathologists whereas GPT-3.5 was equally good as the human members involved. Both GPTs posted total scores in the acceptable range of a trainee in a licensing test. It is important to note that the readers could easily classify the responses produced by GPT-3.5 as correct in terms of all questions except one. The results showed that ChatGPT, especially GPT-4, had the ability to respond to questions related to pathology on a similar level of competence, even higher, than trained pathologists. It highlights how large language models can transform the profession of pathology.

Nyqvist et al. [36] assessed the potential of ChatGPT-4 in the construction of projects risk management and compared the work of this AI with that of human professionals. The research also used a mixed-method design to include anonymous peer ratings of both ChatGPT-4 and 16 human risk management experts of Finnish construction firms. The analysis was based on major points of risk management such as the identification of risks, their analysis, and control and integrated both qualitative and quantitative analysis. The results showed that ChatGPT-4 was found to surpass human abilities in the generation of detailed risk management plans as the quantitative scores were much higher than those of the human average. Nevertheless, the limitations to the model were also critical as the model had weaknesses in areas of specificity and practicality where human experience was still the best. This highlights the possibilities of AI to supplement human work and highlights the importance of human judgment in multifaceted and situation-specific work.

Padovan et al. [37], explored the opportunities and shortcomings of artificial intelligence in occupational health care by evaluating the accuracy and reliability of ChatGPT to provide complex medical responses to occupational health-related queries. The research was conducted by a group of physicians who were to develop a set of questions and answers that would revolve around the Italian occupational medicine laws. Doctors had their different areas of subjects so that there was total coverage. Responses to all topics were generated using ChatGPT both with and without references to texts on legislation. In order to have an impartial evaluation, two groups of physicians, who were also blind to the responses, rated the other group in the evaluation process. The findings showed that the occupational physicians performed better than ChatGPT in the ability to develop relevant questions based on a 5-point Likert scale. Nevertheless, the answers of ChatGPT, when it was equipped with access to legislative texts, were equal to those created by licensed medical professionals. This points to the possible usefulness of ChatGPT in situations where legislation needs to be provided, and to the need to have a skilled human involved in the more subtle details of occupational health care.

3 Design & Methods

The main goal of this work is to assess the hypothesis that AI system, ChatGPT, has reached the stage at which it could satisfy the quality criteria of the programming code written by the human professionals. This is done by developing a systematic research methodology as shown in Figure 1.

The research commenced by choosing the computational algorithms. The number of algorithms identified was 367 with 300 being selected to be used in the study. I chose the subjects of the study (selection) on the principles modified in accordance with other significant notable studies [1, 2]. Out of the chosen algorithms, 38 were related to control structures and recursion, 30 to sorting and searching, 38 to matrices, 35 to classical computational methods, 38 to string manipulation, 40 to number theory, 25 to set theory, 25 to combinatorial algorithms, and the rest 25 to geometric algorithms.

After the algorithms had been selected, code generation was done. Two different but similar code implementations of each of the chosen algorithms were produced and compiled into study-specific code corpus. ChatGPT (GPT-4) produced the first set of code in the period between August 18, 2024, and August 28, 2024, whereas the second set was written by a group of three professional programmers. Python is the programming language that

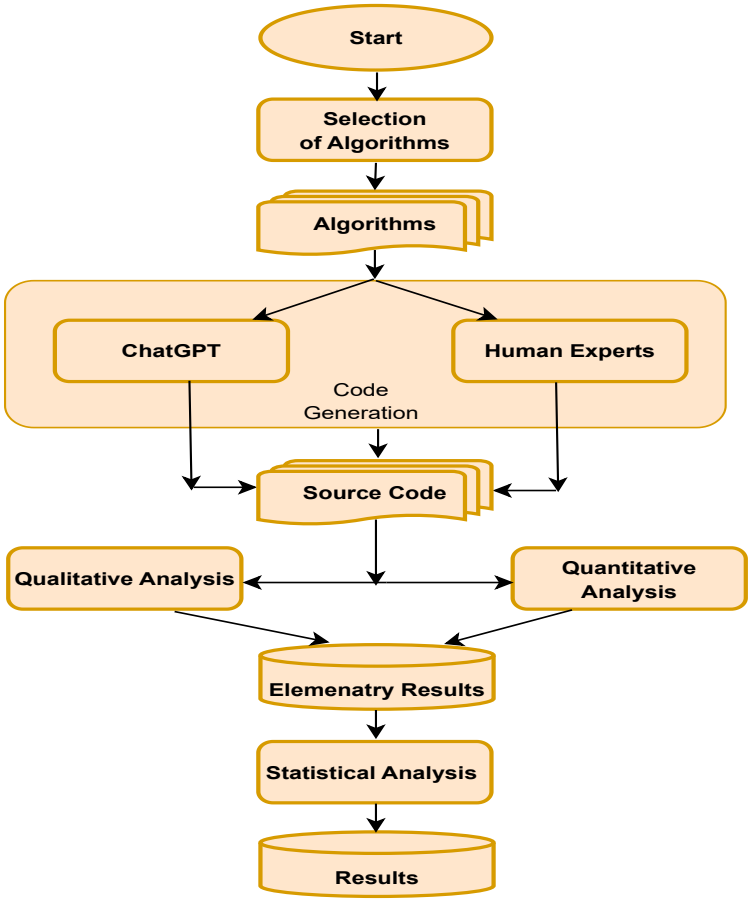

Figure 1. Research Methodology

was selected because of its wide applicability and functionality. Following the production of the code, quantitative and qualitative assessment was performed on both sets.

The quantitative analysis used Halstead Complexity Metrics (HCM) and Cyclomatic Complex (CC), which have been generally perceived to be effective in measuring the complexity of code. Specifically, HCM provided specific information about the structural complexity of the code. Halstead Complexity Metrics are specifically created to measure the complexity of the program, and evaluate the use of source code by analyzing the overall utilization of operands and operators as the basic units of information. The four variables that are fundamental to operators and operands make up these metrics and are as follows [38].

η_1 = Number of (distinct) unique operators
η_2 = Number of (distinct) unique operands
N_1 = Total number of operators
N_2 = Total number of operands

The Halstead Complexity Measures provide a number of different measures to determine code complexity. The Volume, Difficulty and Effort measurements were used in this research to compare the code written by Chat-GPT and human experts.The volume indicates the volume of information the program contains, and how big an algorithmic implementation.The following formula is used to calculate the program's volume.

Volume = $(N_1 + N_2) \times \log_2(\eta_1 + \eta_2)$

The complexity of a program is also directly associated with the number of different operators in a program which is a measure of the complexity of writing or reading the program. The formula below is used to compute the difficulty.

Difficulty = $\left(\frac{\eta_1}{2}\right) \times \left(\frac{N_2}{\eta_2}\right)$

The effort is used to measure the mental effort needed to translate a certain algorithm into a particular programming language. The formula used for calculating effort is as follows.

Effort = $\left(\frac{\eta_1}{2}\right) \times \left(\frac{N_2}{\eta_2}\right) \times \left((N_1 + N_2) \times \log_2(\eta_1 + \eta_2)\right)$

The second metric used in the quantitative analysis was Cyclomatic Complexity, also known as McCabe Complexity, which measures the number of linearly independent paths through a program's code [39]. A higher cyclomatic complexity indicates the need for more test cases to achieve comprehensive code coverage, which often suggests lower code quality.

The second part of the study focused on qualitative analysis, where two human experts, who were not involved in the code generation process, reviewed the code and rated it using a 5-point Likert scale. The experts were not informed which code repository was developed by ChatGPT and which by human experts. Initially, the reviewers individually analyzed the code and then collaborated at a peer level to consolidate their results. For the qualitative analysis, 15 questions (shown in Table 1) were defined to evaluate and rank the programs from both code repositories.

Table 1. Details of Questions for Qualitative Analysis

1	Are the codes readable and clear?
2	Does the code consist of well-documented comments?
3	Does the code consist of well-defined functions?
4	Is the flow of code well-structured logically?
5	Are the codes maintainable?
6	Is the code capable of handling error and exception?
7	Does the code follow any mechanism to handle edge cases?
8	Can the code accommodate larger data inputs?
9	Does the code follow the standard coding convention? (naming conventions)
10	Does the code follow the standard coding convention? (indentation style)
11	Due to its modularity, can the code be used in other applications?
12	Is this code using the most recent programming languages, libraries, and frameworks effectively?
13	Are the chosen tools and technologies appropriate for the task?
14	Are there any bugs or errors in the code that prevent it from functioning correctly?
15	Are the codes directly executable?

The analysis results were statistically examined using SPSS (version 25), while the visualization of the results was conducted using R (version 4.2.3).

4 Results

The study, during the quantitative analysis initially performed the elementary analysis of programs, in which the operators and operands identified and from these program length and program vocabulary were calculated. The results obtained from the elementary analysis are shown in Table 2.

Table 2. Results of Elementary Information in Titles

Attribute	Source	Mean	Median	Std Dev	Range	IQR	Skewness	Kurtosis
Operators	ChatGPT	11.56	8.0	9.62	91	9	3.05	17.17
	Human	9.57	7.0	8.36	64	9	2.34	7.61
Distinct Operators	ChatGPT	3.73	3.0	2.13	11	3	0.99	0.98
	Human	3.32	3.0	2.04	11	2	1.06	1.12
Operands	ChatGPT	107.78	92.0	58.33	439	66	1.92	6.51
	Human	83.41	71.0	49.35	266	67	1.10	1.15
Distinct Operands	ChatGPT	47.97	46.0	16.10	127	20	1.69	7.09
	Human	38.45	36.0	16.74	101	23	0.79	1.05
Program Vocabulary	ChatGPT	51.71	49.0	16.92	133	21	1.71	7.46
	Human	41.76	40.0	17.17	110	23	0.81	1.29
Program Length	ChatGPT	119.34	101.5	65.18	527	72	2.11	8.32
	Human	92.98	84.0	53.75	292	72	1.07	1.06

The study identified a total of 3,467 operators, including 1,120 distinct operators, in the programs generated by ChatGPT. In contrast, the human-crafted programs contained 2,871 operators, with 995 being distinct. Additionally, the ChatGPT-generated programs included 32,335 operands, of which 14,392 were unique, whereas the human-crafted programs consisted of 25,023 operands, with 11,534 being distinct. To provide a clearer illustration of the results from the elementary analysis, line charts have been generated and are presented in Figure 2.

The statistical data obtained from the preliminary analysis were used to compute the difficulty, effort, and volume metrics as defined by Halstead Complexity Metrics. Additionally, the Cyclomatic Complexity of all programs within both code corpora was calculated. The results of these computations are presented in Table 3.

The Halstead Complexity Metric for volume provides a measure of a program's size based on the number of operands and operators it utilizes. In this study, the volume of programs generated by ChatGPT was calculated

Table 3. Results of Complexity Analysis

Attribute	Source	Mean	Median	Std Dev	Range	IQR	Skewness	Kurtosis
Volume	ChatGPT	693.50	588.5	442.04	3900	460	2.68	13.63
	Human	515.48	445.5	345.67	2055	429	1.28	1.93
Difficulty	ChatGPT	4.45	3.0	3.74	33	4	2.95	15.38
	Human	3.71	3.0	3.13	19	3	2.09	5.64
Effort	ChatGPT	4113.71	2145.0	7136.76	73534	3629	5.60	42.71
	Human	2431.26	1216.0	3770.79	27035	2226	3.97	19.76
Cyclomatic Complexity	ChatGPT	6.02	5.0	3.56	24	3	1.55	3.33
	Human	4.95	4.0	2.94	16	4	1.23	1.51

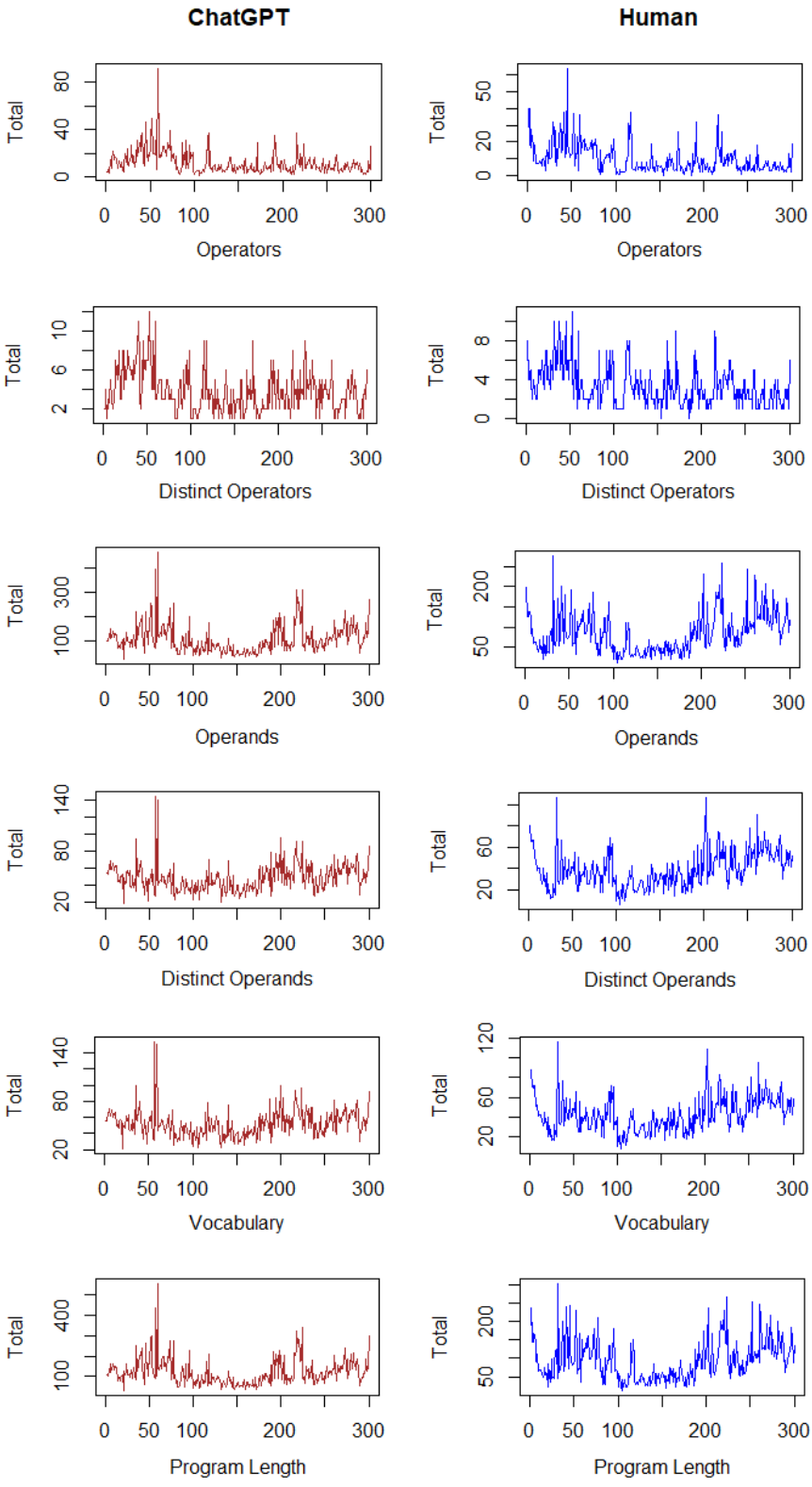

Figure 2. Line Charts

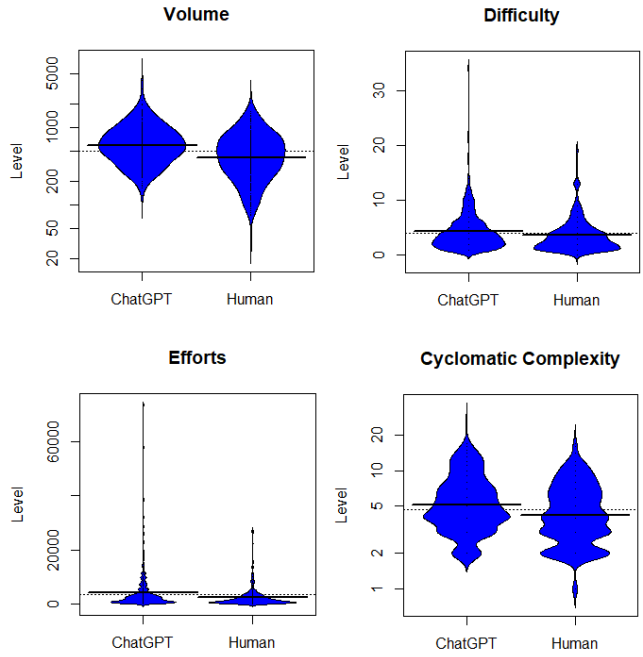

Figure 3. Bean Plots of Complexity Analysis

as 208042.9, compared to 154640.7 for human-crafted programs. Similarly, the cumulative difficulty of ChatGPT-generated programs was found to be 1319.53, whereas the difficulty for human-crafted programs was 1109.55. Difficulty reflects how easily the code can be understood or written.

The cumulative effort required to implement ChatGPT-generated programs was calculated at 1234,114.7, significantly higher than the effort required for human-crafted programs, which stood at 729370.2. Additionally, Cyclomatic Complexity, which measures the control flow and maintainability of programs, was higher for ChatGPT-generated programs at 1805, compared to 1484 for human-generated programs. These differences are visually represented in Figure 3 for better clarity and comparison.

The results of the complex analysis were analyzed through a normality test to estimate the data distribution. The results of these tests are provided in Table 4.

The normality tests showed that the complexities scores were not normally distributed. Hence, non-parametric Mann-Whitney U test further statistical analysis was conducted.

Mann-Whitney U test was then used to analyze the complexity scores which indicated that there were significant scores between the complexity that human experts generated and that of ChatGPT. The results were as follows: volume ($U = 32089$; $Z = -6.081$; $p < 0.05$), difficulty ($U = 38541$; $Z = -3.077$; $p < 0.05$), effort ($U = 34855$; $Z = -4.778$; $p < 0.05$), and Cyclomatic complexity ($U = 36200$; $Z = -4.182$; $p < 0.05$).

Besides the quantitative study, the research also performed a qualitative evaluation of programs created by both chatbots one created by ChatGPT and the other one created by human experts. The following analysis was conducted on the basis of the factors provided in Table 1, with the resulting evaluations presented in Table 5.

The outcomes of the qualitative study were further evaluated in relation to the MannU Whitney test that showed that there were significant differences between ChatGPT-generated code and human-written code in various aspects. Regarding the readability and clarity (Q1), the code produced by ChatGPT received a significantly higher rating, and the mean rating of the produced code (204.21) was lower than that of the human-produced code (396.79), which exceeds $p< 0.05$, meaning that AI-generated code is seen as less difficult to read.

Table 4. Normality Tests on Complexity Scores

Metrics	Source	Kolmogorov-Smirnov			Shapiro-Wilk		
		Statistic	df	Sig.	Statistic	df	Sig.
Volume	ChatGPT	.123	300	< .05	.803	300	< .05
	Human	.116	300	< .05	.902	300	< .05
Difficulty	ChatGPT	.194	300	< .05	.752	300	< .05
	Human	.197	300	< .05	.787	300	< .05
Effort	ChatGPT	.288	300	< .05	.488	300	< .05
	Human	.260	300	< .05	.568	300	< .05
Cyclomatic Complexity	ChatGPT	.189	300	< .05	.854	300	< .05
	Human	.196	300	< .05	.879	300	< .05

Conversely, comment documentation (Q2) was superior in human-written code, showcasing more comprehensive and well-documented comments, as reflected by mean ranks of 184.02 (ChatGPT) versus 416.99 (human), $p < 0.05$. Similarly, for well-defined functions (Q3), human-crafted code achieved a higher mean rank of 187.18 compared to 413.82 for ChatGPT ($p < 0.05$).

When assessing logical structure (Q4), human-written code marginally outperformed ChatGPT-generated code, with a statistically significant mean rank difference of 279.19 versus 321.81 ($p=0.001$). Human-written programs were also significantly more maintainable (Q5), with a mean rank of 227.56 compared to ChatGPT's 373.44.

In the area of error handling capabilities (Q6), human programs were rated superior, with mean ranks of 279.76 (human) versus 321.24 (ChatGPT). Moreover, for handling invalid inputs and empty datasets (Q7), human-crafted programs demonstrated better performance, achieving a mean rank of 330.88 compared to ChatGPT's 269.23 ($p < 0.05$).

Regarding the size of data inputs managed (Q8), human-written code significantly outperformed AI-generated code ($U = 3189.0$, $p < 0.05$). Human programs also adhered more closely to conventional coding standards in naming conventions (Q9) and indentation style (Q10), with U values of 26620.0 and 32997.5, respectively ($p < 0.05$ for both).

Interestingly, modularity and reusability (Q11) showed no significant differences between AI and human-written code ($p = 0.346$). However, human programs were rated better for integration of modern libraries and frameworks (Q12) ($p < 0.05$) and use of relevant tools and technologies (Q13) ($p = 0.002$).

Finally, for bugs and errors (Q14), AI-generated code contained significantly more issues, while human-crafted programs were rated as more directly executable (Q15). The human code demonstrated greater reliability and executability, with $U = 33776.0$, $p < 0.05$ for Q14 and $U = 40397.0$, $p = 0.018$ for Q15. These findings underscore the superior quality and reliability of human-written code across critical dimensions.

5 Discussion

This study conducted a comprehensive comparative analysis of ChatGPT-generated programs and human-crafted programs using Halstead complexity metrics, Cyclomatic complexity, and qualitative assessments. The findings reveal notable differences between the two approaches in terms of complexity, readability, maintainability, and other coding attributes.

The data shows that ChatGPT-generated programs comprise a total of 3467 operators and 32335 operands,

Table 5. Result of Qualitative Analysis

Questions	ChatGPT					Human Experts				
	SD	D	N	A	SA	SD	D	N	A	SA
1	8	177	33	33	49	3	11	11	126	149
2	0	11	9	87	193	5	119	107	45	24
3	1	10	9	85	195	5	122	92	54	27
4	0	14	48	136	102	0	19	72	137	72
5	0	5	7	97	191	1	36	57	136	70
6	0	6	8	138	148	0	6	9	180	105
7	4	86	6	99	105	0	9	7	168	116
8	0	12	7	134	147	12	51	14	127	96
9	3	9	7	140	141	10	101	11	108	70
10	5	84	10	104	97	0	10	18	135	137
11	0	22	12	130	136	0	23	12	142	123
12	7	11	15	134	133	18	53	19	126	84
13	3	71	6	115	105	0	13	10	172	105
14	16	41	13	124	106	32	85	11	111	61
15	14	48	9	114	115	3	5	15	164	113

SD = Strongly Disagree, D = Disagree, N= Neither Agree nor Disagree, A = Agree, SA = Strongly Agree

compared to 2871 operators and 25023 operands in human-generated programs. This represents a 20.8% increase in operators and a 29.2% increase in operands for AI-generated code. The higher average of operators and operands in AI-generated programs indicates that these codes are not only longer but also involve more components to achieve similar functionality as human-written programs.

ChatGPT-generated programs comprise 1120 programs, relative to 995 programs in human-written programs - a 12.6% improvement. Likewise, on individual operands, ChatGPT programs use 14392 different operands, where 11534 different operands appear in human-written code, a difference of 24.8%. These measurements readily indicate that AI-generated code is more complex in its operations.

The amount of distinct operators and operands (program vocabulary) is also greater in AI-generated programs (15512) than in human-generated programs (12529), which is an increase of 23.8%. Similarly, length of the program, a sum of total operators and operands, is 35802 with ChatGPT-generated code and 27894 with human-written code, 28.4% higher. These findings underscore the fact that AI-generated programs are more complex and wordy.

The volume metric, a measure of size of the code, is much larger in the case of AI-generated programs, 264473.28 versus 206752.55 in the case of human written code, which is an increase of 27.9%. This implies that code which is generated by AI is bigger and has more information. This fact is supported by the Mann-Whitney U test with the ratio of the mean ranks of the program generated by AI and the program generated by human being as 343.54 and 257.46 respectively (U = 32089, Z -6.081Z = -6.081, p textless 0.001 p textless 0.001).

Another important measure, difficulty, is also larger in AI-generated code (1258.18) than in human-created

programs (1078.65), where it is greater by 16.6%. This is an indication of higher complexity and lower comprehensibility of AI-generated code, which is more difficult to debug and maintain. This finding is further confirmed by the MannWhitneyU test with mean ranks of 322.03 and 278.97 of AI-generated programs and human-written code respectively (U = 38541, Z= -3.077 p= 0.002).

Effort metric, the effort that has to be consumed in implementation, is significantly greater in ChatGPT-generated programs, 49.3% higher than the effort to write human code. According to the Mann-Whitney U-test, significant differences were detected (U =34855, Z= -4.778, $p < 0.005$), where the mean of the ranks of AI-generated code was 334.32, and the mean of the ranks of human-written code was 266.68.

The cyclomatic complexity, a control path metric of the program, is also greater in code generated by AI-generated code. The average rank of ChatGPT programs is 329.83, as compared to the human-written programs of 271.17. This difference is confirmed by the MannWhitney U test (U = 36200, Z =-4.182, $p <0.005$), which means that AI-generated code contains more complex control structures, which makes the testing and maintenance difficult.

Qualitative aspects of quality of code were also investigated in the study. Ai-generated programs received better scores in the readability and clarity category (Q1), which could be explained by the order of the language models on large datasets. The programs that were written by humans were however better at documenting comments (Q2) and organizing functions (Q3) since human developers usually write comprehensive comments and plan functions more carefully to improve the understanding and maintainability of their code.

Human programs were better in maintainability (Q5) and error handling (Q6) opportunities, as they were robust and stored their flexibility to unpredicted situations. Also, the human-written code performed better in terms of edge cases (Q7) because AI-based models are highly dependent on the input patterns and might not be able to address unusual cases.

Although AI-based programs proved to be more robust when the input data size is large (Q8), human developers were more loyal to the codes (Q9, Q10), identifying the names and the use of indents. Both AI and human code were equally reusable (Q11), yet human programmers were more precise in the choice of tools and technologies (Q12, Q13).

It is worth noting that AI-generated programs had more bugs and errors (Q14) and were less likely to be directly executable (Q15). This makes it clear that much review and testing of AI-generated code is required, especially when it comes to critical applications.

These results highlight the fact that although AI-generated programs can quickly generate large and functional code, they are more verbose, complex, and require more resources than human-written code. These findings are of great importance to the software development, education and industry practices.

The paper also emphasizes the importance of the organizations to take into account the complexities of AI-generated codes when scheduling, addressing resources, and introducing AI tools to their workflows. This research is a unique addition to the knowledge base on the quantitative and qualitative differences between code generated by AI and human beings with the help of Halstead metrics and cyclomatic complexity measurements.

The findings of the study are founded on particular datasets and program exercises, which can restrict the possibility of generalization. As a growing field, future studies would be able to increase the breadth of research by adding other tasks and areas. Moreover, it should be aimed at enhancing the maintainability and efficiency of AI-generated code so that it could be successfully used in a variety of applications.

6 Conclusion

The fast advancement of the computing industry and more so artificial intelligence has transformed the world with regards to societies, organizations and industries. ChatGPT and other large language models have man-

aged to execute a large number of tasks in many different areas with incredible efficiency and efficiency, which proves to be incredibly versatile. Although these new products have come with potent automation and problem solving tools, there have been concerns regarding the risk of job replacement especially in technical areas like programming. This paper give a critical comparison of the code generated by ChatGPT and the human programmer with regard to its complexity, readability, maintainability, errors, and adherence to coding standards. The results suggest that although ChatGPT is capable of generating readable and functional code at a high rate, the human developers retain an advantage regarding the aspects that influence the sustainability of the quality of the software. Specifically, human-written code out-performs AI-written code on such metrics as maintenance ease, appropriate handling and detection of errors, and compliance with industrial standards. The results show that AI-generated code is full of opportunities, but it generally requires much verification and refinement before it can be implemented in a real-world context, particularly in high-stakes environments where reliability is vital. Thus, ChatGPT assists with the process of automation of the code, yet it is still not capable of replacing human coders entirely. To sum up, this paper has reported on the recent developments in artificial intelligence (ChatGPT) as empowerment tools that can make people, and society in general more productive, creative, and benefited instead of being a threat. AI is supposed to be viewed as a collaborative co-worker that enhances and not substitutes the human abilities, pushing the software development and improvement in other areas. AI is to be regarded as a collaborative organization that can complement human abilities and promote software development and other areas.

Author Contributions

Samina Azeem: Conceptualization, Methodology, Analysis, Writing- Reviewing and Editing. **Muhammad Shumail Naved**: Supervision, Statistical Analysis, Visualization. **Muhammad Sajid**: Data Collection. **Imran Ali**: Validation.

Compliance with Ethical Standards

The authors declare no conflict of interest in this study. The human evaluators participated in the study voluntarily with informed consent and the data are anonymized. All the data are available upon appropriate request. This study received no external funding. The results were reported without any manipulation of the data or misrepresentation.

References

[1] M. S. Naveed, “Measuring the programming complexity of c and c++ using halstead metrics,” *University of Sindh Journal of Information and Communication Technology*, vol. 5, no. 4, pp. 2521–5582, 2021.

[2] M. S. Naveed, “‘comparison of c++ and java in implementing introductory programming algorithms,” *QUEST Research Journal*, vol. 19, no. 1, pp. 95–103, 2021.

[3] K. Wilson, “Introduction to computer programming,” in *The Absolute Beginner’s Guide to Python Programming: A Step-by-Step Guide with Examples and Lab Exercises*, pp. 1–13, Springer, 2022.

[4] M. Shoaib, M. S. Naveed, A. A. Sanjrani, A. Ahmed, *et al.*, “A comparative study of contemporary programming languages in implementation of classical algorithms,” *Journal of Information & Communication Technology (JICT)*, vol. 14, no. 1, 2021.

[5] P. Li, “The research for software programming of english online learning,” in *2022 IEEE 2nd International Conference on Electronic Technology, Communication and Information (ICETCI)*, pp. 707–710, IEEE, 2022.

[6] L. A. Kumar and D. K. Renuka, *Deep learning approach for natural language processing, speech, and computer vision: techniques and use cases*. CRC Press, 2023.

[7] G. Franceschelli and M. Musolesi, "Creativity and machine learning: A survey," *ACM Computing Surveys*, vol. 56, no. 11, pp. 1–41, 2024.

[8] M. S. Naveed, "Quantifying similarities: Oncology documents from google bard and chatgpt," *International Journal of Innovations in Science & Technology*, vol. 5, no. 4, pp. 773–786, 2023.

[9] Y. K. Dwivedi, L. Hughes, E. Ismagilova, G. Aarts, C. Coombs, T. Crick, Y. Duan, R. Dwivedi, J. Edwards, A. Eirug, *et al.*, "Artificial intelligence (ai): Multidisciplinary perspectives on emerging challenges, opportunities, and agenda for research, practice and policy," *International journal of information management*, vol. 57, p. 101994, 2021.

[10] S. Maleki Varnosfaderani and M. Forouzanfar, "The role of ai in hospitals and clinics: transforming healthcare in the 21st century," *Bioengineering*, vol. 11, no. 4, p. 337, 2024.

[11] S. Kolluri, J. Lin, R. Liu, Y. Zhang, and W. Zhang, "Machine learning and artificial intelligence in pharmaceutical research and development: a review," *The AAPS journal*, vol. 24, pp. 1–10, 2022.

[12] G. Albahri, A. A. Alyamani, A. Badran, A. Hijazi, M. Nasser, M. Maresca, and E. Baydoun, "Enhancing essential grains yield for sustainable food security and bio-safe agriculture through latest innovative approaches," *Agronomy*, vol. 13, no. 7, p. 1709, 2023.

[13] J. C. Tellez Gaytan, K. Ateeq, A. Rafiuddin, H. M. Alzoubi, T. M. Ghazal, T. A. Ahanger, S. Chaudhary, and G. Viju, "Ai-based prediction of capital structure: Performance comparison of ann svm and lr models," *Computational intelligence and neuroscience*, vol. 2022, no. 1, p. 8334927, 2022.

[14] L. Chen, P. Chen, and Z. Lin, "Artificial intelligence in education: A review," *Ieee Access*, vol. 8, pp. 75264–75278, 2020.

[15] A. A. Khan, A. A. Laghari, A. M. Baqasah, R. Bacarra, R. Alroobaea, M. Alsafyani, and J. A. J. Alsayaydeh, "Bdlt-iomt—a novel architecture: Svm machine learning for robust and secure data processing in internet of medical things with blockchain cybersecurity," *The Journal of Supercomputing*, vol. 81, no. 1, pp. 1–22, 2025.

[16] A. A. Khan, J. Yang, A. A. Laghari, A. M. Baqasah, R. Alroobaea, C. S. Ku, R. Alizadehsani, U. R. Acharya, and L. Y. Por, "Baiot-ems: Consortium network for small-medium enterprises management system with blockchain and augmented intelligence of things," *Engineering Applications of Artificial Intelligence*, vol. 141, p. 109838, 2025.

[17] A. A. Khan, A. A. Laghari, A. M. Baqasah, R. Alroobaea, A. Almadhor, G. A. Sampedro, and N. Kryvinska, "Blockchain-enabled infrastructural security solution for serverless consortium fog and edge computing," *PeerJ Computer Science*, vol. 10, p. e1933, 2024.

[18] A. A. Khan, Y.-L. Chen, F. Hajjej, A. A. Shaikh, J. Yang, C. S. Ku, and L. Y. Por, "Digital forensics for the socio-cyber world (df-scw): A novel framework for deepfake multimedia investigation on social media platforms," *Egyptian Informatics Journal*, vol. 27, p. 100502, 2024.

[19] H. H. Rashidi, J. Pantanowitz, M. Hanna, A. P. Tafti, P. Sanghani, A. Buchinsky, B. Fennell, M. Deebajah, S. Wheeler, T. Pearce, *et al.*, "Introduction to artificial intelligence (ai) and machine learning (ml) in pathology & medicine: generative & non-generative ai basics," *Modern Pathology*, p. 100688, 2025.

[20] H. Javed, S. El-Sappagh, and T. Abuhmed, "Robustness in deep learning models for medical diagnostics: security and adversarial challenges towards robust ai applications," *Artificial Intelligence Review*, vol. 58, no. 1, pp. 1–107, 2025.

[21] J. Segessenmann, T. Stadelmann, A. Davison, and O. Dürr, "Assessing deep learning: a work program for the humanities in the age of artificial intelligence," *AI and Ethics*, vol. 5, no. 1, pp. 1–32, 2025.

[22] S. Shoukat, T. Gao, D. Javeed, M. S. Saeed, and M. Adil, "Trust my ids: An explainable ai integrated deep learning-based transparent threat detection system for industrial networks," *Computers & Security*, vol. 149, p. 104191, 2025.

[23] L. Albshaier, S. Almarri, and A. Albuali, "Federated learning for cloud and edge security: A systematic review of challenges and ai opportunities," *Electronics*, vol. 14, no. 5, p. 1019, 2025.

[24] D. Thakur, A. Guzzo, G. Fortino, and F. Piccialli, "Green federated learning: A new era of green aware ai," *ACM Computing Surveys*, 2025.

[25] A. Bucaioni, H. Ekedahl, V. Helander, and P. T. Nguyen, "Programming with chatgpt: How far can we go?," *Machine Learning with Applications*, vol. 15, p. 100526, 2024.

[26] R. Jain, J. Thanvi, and A. Subasinghe, "The evolution of chatgpt for programming: a comparative study," *Engineering Research Express*, 2025.

[27] A. Koubaa, B. Qureshi, A. Ammar, Z. Khan, W. Boulila, and L. Ghouti, "Humans are still better than chatgpt: Case of the ieeextreme competition," *Heliyon*, vol. 9, no. 11, p. e21624, 2023.

[28] R. Yilmaz and F. G. K. Yilmaz, "Augmented intelligence in programming learning: Examining student views on the use of chatgpt for programming learning," *Computers in Human Behavior: Artificial Humans*, vol. 1, no. 2, p. 100005, 2023.

[29] L. Grundner and B. Neuhofer, "The bright and dark sides of artificial intelligence: A futures perspective on tourist destination experiences," *Journal of Destination Marketing & Management*, vol. 19, p. 100511, 2021.

[30] V. Taecharungroj, ""what can chatgpt do?" analyzing early reactions to the innovative ai chatbot on twitter," *Big Data and Cognitive Computing*, vol. 7, no. 1, p. 35, 2023.

[31] O. Temsah, S. A. Khan, Y. Chaiah, A. Senjab, K. Alhasan, A. Jamal, F. Aljamaan, K. H. Malki, R. Halwani, J. A. Al-Tawfiq, *et al.*, "Overview of early chatgpt's presence in medical literature: insights from a hybrid literature review by chatgpt and human experts," *Cureus*, vol. 15, no. 4, 2023.

[32] J. Steiss, T. Tate, S. Graham, J. Cruz, M. Hebert, J. Wang, Y. Moon, W. Tseng, M. Warschauer, and C. B. Olson, "Comparing the quality of human and chatgpt feedback of students' writing," *Learning and Instruction*, vol. 91, p. 101894, 2024.

[33] D. Duong and B. D. Solomon, "Analysis of large-language model versus human performance for genetics questions," *European Journal of Human Genetics*, vol. 32, pp. 466–468, 2024.

[34] F. Breithaupt, E. Otenen, D. R. Wright, J. K. Kruschke, Y. Li, and Y. Tan, "Humans create more novelty than chatgpt when asked to retell a story," *Scientific Reports*, vol. 14, p. 875, 2024.

[35] A. Y. Wang, S. Lin, C. Tran, R. J. Homer, D. Wilsdon, J. C. Walsh, E. A. Goebel, I. Sansano, S. Sonawane, V. Cockenpot, *et al.*, "Assessment of pathology domain-specific knowledge of chatgpt and comparison to human performance," *Archives of pathology & laboratory medicine*, vol. 148, no. 10, p. 1152–1158, 2024.

[36] R. Nyqvist, A. Peltokorpi, and O. Seppänen, "Can chatgpt exceed humans in construction project risk management?," *Engineering, Construction and Architectural Management*, vol. 31, no. 13, pp. 223–243, 2024.

[37] M. Padovan, B. Cosci, A. Petillo, G. Nerli, F. Porciatti, S. Scarinci, F. Carlucci, L. Dell'Amico, N. Meliani, G. Necciari, *et al.*, "Chatgpt in occupational medicine: a comparative study with human experts," *Bioengineering*, vol. 11, no. 1, p. 57, 2024.

[38] M. S. Naveed, "Pedagogical suitability: A software metrics-based analysis of java and python," *International Journal of Innovations in Science Technology*, vol. 6, no. 4, pp. 1956–1967, 2024.

[39] A. Odeh, M. Odeh, N. Odeh, and H. Odeh, "Machine learning model for measuring cyclomatic complexity of source code," in *2023 International Conference on Intelligent Computing, Communication, Networking and Services (ICCNS)*, pp. 149–153, IEEE, 2023.