Uluslararası İleri Doğa Bilimleri ve Mühendislik Araştırmaları Dergisi
Sayı 8, S. 445-464, 11, 2024

Araştırma Makalesi

IJANSER
https://as-proceeding.com/index.php/ijanser
ISSN:2980-0811

International Journal of Advanced Natural Sciences and Engineering Researches
Volume 8, pp. 445-464, 11, 2024

Research Article

Generative Artificial Intelligence in Computer Programming Education: A Bibliometric Analysis

Turgay DEMİREL[*1]

[1]*Information Systems and Technologies Department, Faculty of Applied Sciences, Ataturk University, Türkiye*

[*]*(tdemirel@atauni.edu.tr)*

(Received: 12 December 2024, Accepted: 17 December 2024)

(4th International Conference on Frontiers in Academic Research ICFAR 2024, December 13-14, 2024)

ATIF/REFERENCE: Demirel, T. (2024). Generative Artificial Intelligence in Computer Programming Education: A Bibliometric Analysis. *International Journal of Advanced Natural Sciences and Engineering Researches*, 8(11), 445-464.

Abstract **–** This study aims to investigate the emerging trends and research patterns surrounding generative AI in computer programming education through bibliometric analysis. It seeks to identify descriptive characteristics of existing literature, influential research contributions, core terms, and future research directions. A bibliometric analysis was conducted using data from the Scopus database, focusing on publications related to generative AI in programming education. The analysis employed techniques such as citation analysis, co-author analysis, and keyword co-occurrence, utilizing R Studio and VOSviewer for data visualization. The analysis revealed 48 publications, predominantly research articles, with a significant increase in output during 2023-2024. Key journals identified include Applied Sciences and Computers and Education: Artificial Intelligence. Influential authors and countries were highlighted, with China and the USA leading in scientific production. Core terms included "programming education," "artificial intelligence," and "ChatGPT." The thematic map analysis identified motor themes such as "Programming" and "AI in Education," alongside emerging themes like "computational thinking," indicating areas requiring further exploration. This study contributes to the understanding of generative AI's role in programming education by mapping existing research and identifying gaps. The findings from this bibliometric analysis provide a basis for directing future research and cultivating a thorough understanding of how generative AI is shaping the future of computer programming education.

Keywords – Generative Artificial Intelligence, Generative Ai, Computer Programming Education, Large Language Models Bibliometric Analysis.

I. INTRODUCTION

Generative AI is a transformative technology that can create novel content by analyzing complex patterns in extensive datasets using deep learning models. This innovative capability extends across diverse domains, including text, images, audio, video, and code, fostering creativity and innovation in fields like education, art, and research ([1], [2], [3]).

Generative AI tools like ChatGPT and GitHub Copilot are revolutionizing programming education by enhancing software development through real-time code suggestions, error detection, and intelligent completions. These tools can significantly boost developer productivity and learning outcomes ([4], [5], [6]). However, integrating generative AI into programming education raises critical concerns regarding accuracy, context, and ethics, necessitating a careful examination of its impact on student engagement, motivation, and skill development ([7], [8]).

While generative AI tools offer potential benefits, over-reliance can lead to academic dishonesty and shallow understanding of programming for novice learners. Therefore, educators must find a balance between leveraging AI advantages and ensuring students deeply engage with programming principles ([9], [10]).

The emergence of generative artificial intelligence in recent years and its widespread adoption have led to a growing body of research on its applications in various fields, including computer programming education. Although this research area is relatively new, there has been a noticeable increase in the number of studies conducted in recent years. Examining the trends and developments within this research landscape is crucial in informing and guiding future studies in this domain.

The rise of generative artificial intelligence and its widespread adoption in recent years have sparked a growing body of research exploring its applications across diverse fields, including computer programming education. While this research domain is relatively novel, there has been a notable surge in the number of studies conducted in recent times. Examining the trends and advancements within this research landscape is pivotal in informing and guiding future investigations in this domain.

This study aims to explore the emerging trends and research landscape surrounding generative AI in computer programming education through bibliometric analysis. By investigating the characteristics of existing literature, influential research, core terms, and future research directions, this study seeks to provide a comprehensive future research direction.

A. Literature Review

Generative AI is a form of artificial intelligence that can generate unique content. It analyzes patterns and structures in existing data, then leverages this understanding to generate similar yet unique outputs. In contrast to traditional AI systems that categorize or forecast based on input, generative AI produces something novel. Specifically, generative AI utilizes deep learning models to accomplish this content creation process. The models are trained on huge datasets, enabling them to detect complex patterns and structures within the data. The model generates outputs that resemble the training data in style and content, without simply copying it. A generative AI model can create new, realistic images that it has not encountered before ([1], [2], [3]). Generative AI can produce various contents, including text, images, audio, video, and computer code. It can be used to create articles, poems, scripts, translations, realistic photos, artwork, design mockups, music, sound effects, and software programs ([1], [3]). The development of generative AI has introduced captivating opportunities for creativity and innovation in multiple disciplines, encompassing education, art, and scientific endeavors ([11], [12]).

Generative AI is rapidly transforming many aspects of education, including computer programming education. Several recent papers discuss the opportunities and challenges presented by generative AI tools like ChatGPT, CoPilot and others ([4]-[6], [13]-[17]).

Generative AI tools have significantly advanced programming, automating and enhancing software development. Tools like GitHub Copilot and OpenAI Codex provide real-time code suggestions, error detection, and intelligent completions, improving productivity. However, these tools also face accuracy, contextual, and ethical challenges. Generative AI tools can automate a range of programming tasks, including generating code snippets, offering context-aware recommendations, and predicting code completions. This can improve developer productivity by speeding up coding and reducing errors ([7], [8]). Additionally, AI can analyze code and explain its functionality, helping developers better understand complex codebases ([18], [19]).

Generative AI tools can assist programming in various ways, such as simplifying code, translating between languages, refactoring to improve quality, identifying and correcting bugs, and providing personalized feedback and guidance ([5], [18], [20]-[23]). However, translations may not be perfect, and human validation is often needed for code refactoring. AI tools can benefit novice programmers by improving task completion, code quality, and learning efficiency. They can also make complex programming concepts more accessible, which can boost student engagement and motivation ([12], [20]).

Generative AI tools can enhance students' learning by providing personalized guidance and explanations. Studies show students who use AI to understand programming concepts tend to have better learning outcomes ([24], [25]). These tools can also boost students' confidence and self-assurance, as seen with agriculture students using ChatGPT for microcontroller programming [26]. Generative AI can further aid novice programmers by offering hints and explanations to help them overcome challenges and improve problem-solving [27].

Generative AI tools can aid programming tasks, but their usefulness for novice learners varies [5]. These tools may cause misunderstandings and increased cognitive load, especially for complex problems. While effective for simpler tasks, generative AI struggles with more complex issues, requiring additional effort for accurate solutions [2]. Institutions have differing views on integrating these tools, with some banning and others encouraging use with caution [28].

However, there are crucial factors that must be considered when integrating generative AI into computer programming education. Overreliance on generative AI tools in programming education can lead to concerning issues. Students may become overly dependent on these tools, using them to complete assignments without fully grasping underlying concepts. This can foster academic dishonesty and undermine the development of essential problem-solving and critical thinking skills ([9], [10], [15]). Additionally, heavy reliance on AI tools may reduce students' tendency to explore other educational resources, and their academic performance may not improve as a result [2]. Research indicates that heavy reliance on AI-generated solutions can reduce students' engagement in problem-solving and critical thinking [2]. This can be especially problematic for high-achieving students, as over-dependence on AI may hinder their learning process [29]. Students may incorrectly judge their programming skills due to the ease of getting AI-generated solutions, leading to an unrealistic confidence that can leave them unprepared for real-world programming challenges [1] .

In conclusion, generative AI represents a transformative force in programming education, offering significant advantages in productivity and learning outcomes. However, its integration must be approached with caution, as overreliance can hinder the development of critical skills and foster academic dishonesty. Educators and institutions must strike a balance, leveraging the benefits of generative AI while ensuring that students engage deeply with programming concepts to cultivate problem-solving abilities and genuine understanding.

B. Significance and Purpose of the Study

Existing research on the impact of AI-powered code generation tools in programming education lacks depth and comprehensiveness. The current literature does not provide a thorough understanding of the educational implications and practical application value of these tools. Educational research lags behind the rapid technical development of AI tools. This shows the need for literature to provide guidance on integrating AI into programming education [14]. Existing research suggests several research gaps that need to be investigated:

Examining how generative AI impacts student engagement, motivation, and learning in programming courses ([14], [30]), developing strategies to promote responsible and ethical use of these tools while addressing over-reliance and misuse [5], investigating ethical concerns like plagiarism, bias, and impact on creativity [14], studying the long-term effects on programming skills, problem-solving, and career paths [31], examining how the tools work for diverse learners [31], exploring how they can aid collaborative and project-based learning [14], and analyzing the novelty effect in programming education [30] are important areas for further research.

Moreover, identifying the trends of previous studies in the field of generative artificial intelligence in computer programming education is also crucial. This relatively new area has garnered increasing attention from researchers in recent years. Examining these trends can shed light for future research and identify potential areas warranting further exploration. The findings from evaluating the scientific activity in this research area can benefit various stakeholders, including researchers, academics, educators, managers, and other interested parties.

This study aims to explore the emerging research trends and patterns surrounding the use of generative AI in computer programming education by examining the following research questions:

RQ1. What are the descriptive features of research on generative AI in programming education?

RQ2. What is the significant research on generative AI in programming education?

RQ3. What are the key terms explored in research on generative AI in programming education?

RQ4. What are the existing research themes and future research directions of research on generative AI in programming education?

II. METHODOLOGY

A. Research Design

Bibliometrics, a subfield of Scientometrics, employs mathematical and statistical techniques to analyze scholarly literature, with the goal of studying and evaluating scientific activities ([32], [33]). Pioneered by E. Garfield in the mid-20th century, bibliometrics has become a widely adopted approach to help revise knowledge across various disciplines [34].

The aim of this study is to investigate and analyze research trends and patterns related to generative AI in programming education. To accomplish this, bibliometric analysis has been conducted which is a reliable technique [35] that examines important aspects and emerging themes within a particular field [36]. Bibliometric analysis visually depicts the connections between central concepts in a research area, helping researchers better comprehend these relationships [37]. Bibliometric analysis has contributed to the review of scientific knowledge and has been used successfully in different scientific fields ([38], [39], [40]).

The study's methodology provided a framework that guided the research questions, with a focus on performance analysis and science mapping [36]. Performance analysis evaluates the contributions made by research entities, while science mapping explores the relationships between these entities [36]. Performance analysis examines various research components commonly found in review studies, including authors, institutions, countries, and journals. This study utilized various science mapping techniques, such as citation analysis, co-citation analysis, bibliographic coupling, co-word analysis, and co-authorship analysis, to explore the research landscape. Specifically, the first and second research questions examined performance metrics related to the use of generative AI in programming education, while the third and fourth questions employed these science mapping approaches.

The researchers employed R-studio software and open-source bibliometrix packages to perform in-depth science mapping and visualizations for the bibliometric analysis [41]. Researchers also used the VOSviewer software tool (version 1.6.20) for bibliometric mapping and analysis due to its reliability and suitability for this purpose. The VOSviewer tool enabled the processing and grouping of keywords to visualize co-occurrence patterns [42].

B. Dataset

Relying on data from multiple databases for bibliometric analysis can yield divergent results [43]. Therefore, choosing the appropriate database is crucial. Scopus was selected as the database for this study due to its recognition as a comprehensive scholarly resource within the academic community [44]. A search was conducted in the Scopus database, leveraging its extensive coverage and reliability to facilitate the bibliometric mapping analysis ([45], [46]). The search was carried out by examining the title, abstract, and keyword fields in the Scopus database, employing the provided search string: (TITLE-ABS-KEY("artificial intelligence" OR "ai" OR "chatgpt" OR "generative ai" OR "generative artificial intelligence") AND TITLE-ABS-KEY("programming education" OR "teaching programming" OR "learning programming" OR "computer science education" OR "computer programming education")).

The initial search yielded 670 records, which were then filtered according to document type (article, review) and language (English) with no timespan limitation. The decision was to include only articles, both in open and unopen access, as they are the only documents that have undergone the peer review process to ensure scientific quality. The search returned 157 articles after refinement. After that, the researchers reviewed the titles and abstracts of all publications, removing those that were not relevant to generative AI in computer programming education. Thus, the final sample of articles analyzed was obtained through a search in December 2024 and included a total of 48 articles.

C. Data Analysis

The researchers compiled and organized the data from the Scopus core collection into BibTeX and CSV file formats. The study employed the user-friendly biblioshiny web interface within RStudio, in combination with the bibliometrix package, to conduct a bibliometric analysis and visualization. This allowed the researchers to generate tables and graphs without the need for extensive coding expertise, leveraging the intuitive interface provided by these tools.

The study's data analysis encompassed four key stages: descriptive, influential, core, and conceptual foundations of this research domain. The initial stage offered an overview of the field, examining its descriptive characteristics and research topics from the perspective of the authors' countries. The second stage involved an analysis of influential factors, such as journals, articles, authors, institutions, and countries that have significantly shaped the research on generative AI in computer programming education. The third stage examined the frequently used keywords, evaluating the knowledge structure through an analysis of keyword plus and author keywords. The fourth stage then explored the field's conceptual foundation by analyzing the relationships between key concepts and conducting a thematic assessment [38].

III. RESULTS AND DISCUSSION

RQ1. Descriptive Features of Research on generative AI in generative AI in computer programming education

The analysis of the Scopus database revealed 48 publications on the use of generative AI in computer programming education literature between 1994 and 2024 (see Table 1). But the vast majority (%87,5) of the studies were conducted in the 2023-2024 time span. The analysis suggests that the field of generative AI in computer programming education is relatively new and demonstrates a growing trend. The findings align with a previous study that examined research trends on ChatGPT in education [38].

Table 1. Descriptive Characteristics of Studies on Generative AI in Computer Programming Education

Description	n	%
Timespan		
2024	34	70.8
2023	8	16.7
1994-2022	6	12.5
Types		
Research article	46	95.8
Review article	2	4.2
Keywords		
Keywords Plus (ID)	222	55,6
Author's Keywords (DE)	177	44.4
Authors		
Total number of authors	153	100
Single-authored	8	16.7
Multi-authored	40	83.3
International co-authorships	9	18.75
Avg. of co-authors per doc	3.29	-

The analysis reveals that 95.8% of the publications are research articles, while the remaining 4.2% are review articles. The publications collectively included 222 keywords plus and 177 author keywords. Each publication had approximately 3 co-authors (Mean = 3.29), with collaborative authorship observed in research papers (83,3%). The analysis indicates that 18.75% of the publications involved international collaboration among authors.

The analysis suggests a growing interest and emphasis on generative AI within the field of computer programming education, particularly in the 2023-2024 timeframe. The predominance of research articles suggests a focus on empirical studies, while the collaborative nature of authorship highlights the interdisciplinary approach needed to advance this innovative field. Future research should further explore international partnerships.

RQ2. Significant Research on generative AI in computer programming education

Influential Sources

The researchers employed Source Impact and Bradford's Law to identify the core journals in this research domain. Table 2 lists the most prominent journals in this research field, ranked based on their g-index, h-index, m-index, total citations (TC), and net production (NP). These metrics are presented in descending order by the g-index, which is an author-level measurement. The three most influential sources in the field are: Applied Sciences (Switzerland), Computers and Education: Artificial Intelligence and ACM Inroads.

Table 2. Source Impact of the Most Influential Journals

Top Influential Sources	h_index	**g_index**	m_index	TC	NP
Applied Sciences (Switzerland)	2	4	0.4	366	4
Computers and Education: Artificial Intelligence	2	4	1	136	4
ACM Inroads	2	2	1	22	2
ACM Computing Surveys	1	1	0.33	25	1
Computers and Education Open	1	1	1	1	1
Decision Sciences Journal of İnnovative Education	1	1	1	1	1
Education and Information Technologies	1	1	1	2	1
Electronic Journal of E-learning	1	1	1	3	1
Engineering Letters	1	1	0.25	69	1
Humanities and Social Sciences Communications	1	1	1	7	1

Table 3 categorizes the top journals in this research domain into three zones based on Bradford's Law, which groups journals by their citation frequency. The analysis identified 5 journals out of the 36 analyzed as core sources in Zone 1, highlighting their significance in the research on generative AI in computer programming education. Moreover, 16 sources were in Zone 2, and 15 were in Zone 3. Applied Sciences (Switzerland) was ranked as the top journal in Zone 1, followed by Computers and Education: Artificial Intelligence in this research domain. The prominence of journals such as Applied Sciences (Switzerland) and Computers and Education: Artificial Intelligence underscores their critical role in advancing scientific literature and innovation in this field.

Table 3. Sources Categorized According to the Bradford’s Law

Sources	**F**	**CF**	**Zone**
Applied Sciences (Switzerland)	4	4	Zone 1
Computers and Education: Artificial Intelligence	4	8	
IEEE Transactions on Education	3	11	
IEEE Transactions on Learning Technologies	3	14	
ACM Inroads	2	16	

Note: F = citation frequency, CF = cumulative citation frequency.

Influential Articles

Table 4 presents the list of the top 10 most influential articles in the selected timeframe. Studies are listed in descending order according to the total number of citations. These studies collectively examine diverse aspects of generative AI and associated technologies within the context of computer programming education.

Table 4. Most Globally Cited Documents

Author (year)	TC	NTC	Title
[47]	318	5.14	ChatGPT for Education and Research: Opportunities, Threats, and Strategies
[48]	131	2.12	The effect of generative artificial intelligence (AI)-based tool use on students' computational thinking skills, programming self-efficacy and motivation
[49]	69	1.00	Python-bot: A chatbot for teaching python programming
[50]	47	1.00	Source Code assessment and classification based on estimated error probability using attentive lstm language model and its application in programming education
[51]	44	1.00	A review of AI-supported tutoring approaches for learning programming
[52]	38	1.00	The impact of a peer-learning agent based on pair programming in a programming course
[53]	25	1.00	A Survey of Automated Programming Hint Generation: The HINTS Framework
[54]	15	7.08	ChatGPT: Challenges and Benefits in Software Programming for Higher Education
[55]	15	0.24	Teaching CS-101 at the Dawn of ChatGPT
[56]	14	6.61	Would ChatGPT-facilitated programming mode impact college students' programming behaviors, performances, and perceptions? An empirical study

Note: TC = total citation, NTC = normalized total citation

After analyzing the titles of these studies, they explore various aspects of incorporating generative AI into computer programming education. The studies examine the integration of AI-based tools in computer programming education, investigating their impact on student learning and engagement. They also explore collaborative learning approaches that leverage these generative AI technologies. Additionally, the publications address the potential challenges and benefits of integrating AI in educational settings, providing a comprehensive understanding of this emerging field. This indicates that the integration of generative AI in computer programming education presents significant opportunities for enhancing student engagement and learning outcomes, while also posing challenges that require careful consideration and strategic implementation

Influential Authors, Affiliations, and Countries

This section examines the key contributors, including influential authors, organizations, institutions, and countries, who have made significant impacts in this research field. As seen in Table 5, the top five influential authors in this field are presented based on their h-index scores.

Table 5. Most Influential Authors

Author	h_index	g_index	m_index	TC	NP
RAHMAN MM	2	2	0.4	365	2
STRICKROTH S	2	2	0.167	48	2
WATANOBE Y	2	2	0.4	365	2
ADE-IBIJOLA A	1	1	0.25	69	1
AZAIZ I	1	1	0.5	4	1

Note: TC=total citation, NP=net production.

Table 6 ranks the most cited and productive countries and organizations in generative AI in the computer programming education literature. China, the USA, Brazil and Germany have emerged as the leading countries in terms of scientific production, while Bangladesh, Turkey, Japan, and Korea appear to be the countries with the highest citation impact. The analysis reveals that Goethe University, Zhejiang University, Southwest University, University of Maribor, and University of Naples "Parthenope" are among the top-ranked affiliations in terms of the number of published articles in this research domain. The findings highlight a significant disparity in scientific production and citation impact among countries in generative AI research within computer programming education. Notably, while China and the USA lead in output, nations like Bangladesh and Turkey excel in citation influence, suggesting diverse research strengths and strategic focuses across global institutions.

Table 6. Influential Countries by Scientific Production and Citations

Scientific Production		**Most Cited Countries**		**Relevant Affiliation**	
Country	**f**	**Country**	**TC**	**Organization**	**Article**
CHINA	22	BANGLADESH	318	GOETHE UNIVERSITY FRANKFURT	8
USA	10	TURKEY	131	ZHEJIANG UNIVERSITY	7
BRAZIL	6	JAPAN	47	SOUTHWEST UNIVERSITY	6
GERMANY	5	KOREA	42	UNIVERSITY OF MARIBOR	6
JAPAN	4	CHINA	27	UNIVERSITY OF NAPLES “PARTHENOPE”	6
NETHERLANDS	4	USA	21	BEIHANG UNIVERSITY	4
PORTUGAL	4	GERMANY	17	EINDHOVEN UNIVERSITY OF TECHNOLOGY	4
SOUTH KOREA	4	BRAZIL	15	ST. PÖLTEN UNIVERSITY OF APPLIED SCIENCES	4
GHANA	3	AUSTRIA	7	THE UNIVERSITY OF AIZU	4
IRAQ	3	UNITED KINGDOM	7	CODE UNIVERSITY OF APPLIED SCIENCES	3

The top 10 countries of corresponding authors in this research field are illustrated in Figure 1. The term "MCP" refers to publications involving collaboration among researchers from multiple countries. In contrast, SCP refers to publications involving researchers from a single country, without international collaboration. For a study to be considered as a MCP, it must have at least one researcher from a country different from the corresponding author's country.

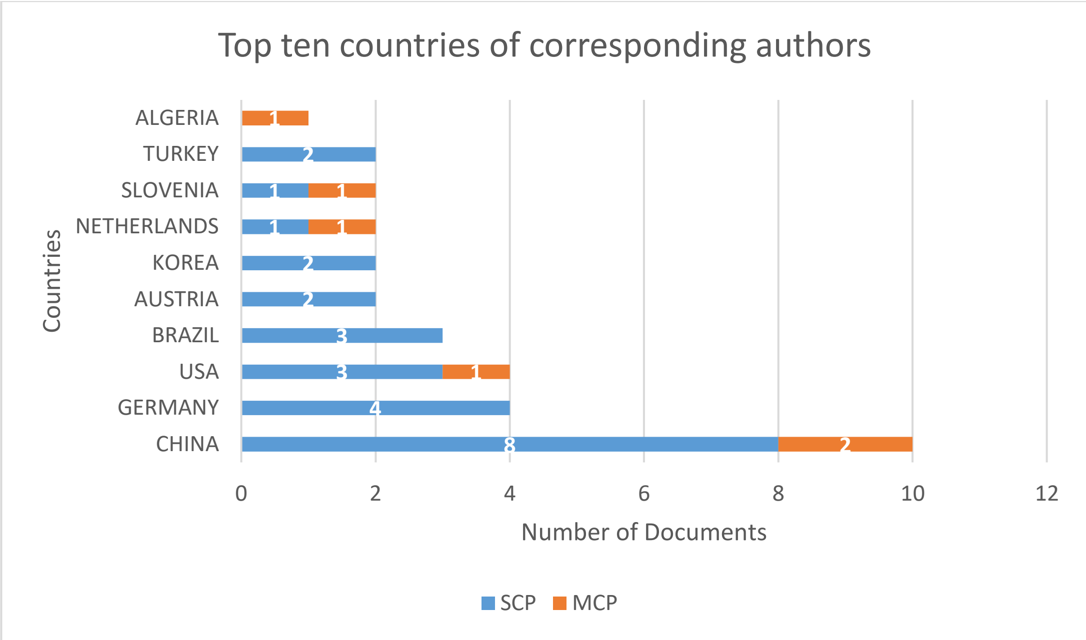

Figure 1. Top Ten Countries of Corresponding Authors

As seen in Figure 1, China is the leading country for corresponding authors, publishing 10 articles with 8 SCPs and 2 MCPs, followed by Germany and USA. Germany has 4 articles (all of them SCPs), and USA has 4 articles (3 SCPs and 1 MCPs). While Japan, Portugal, Ghana, and Iraq are among the top 10 countries in terms of scientific production according to Table 6, they are not represented in the top 10 list of corresponding author countries shown in Figure 1. This suggests that the research contributions from these countries may have different patterns of authorship and collaboration compared to the top corresponding author countries. The analysis reveals that single-country publications (SCPs=77.8%) from leading countries tend to be more prevalent than multi-country publications (MCPs=22.2%) in this research domain. The limited MCPs indicate potential barriers to international collaboration. Further investigation into the factors influencing these trends is needed.

RQ3. Key Terms of Research on generative AI in computer programming education

Table 7 summarizes the top ten most frequently used terms in the literature, encompassing both "keyword plus" and "author's keywords" as the units of analysis. Keyword plus, labeled as IDs, are algorithmically generated from cited reference titles, while author's keywords, labeled as DEs, are directly provided by the researchers [57]. The analysis uses both author keywords and ID keywords, as each provides valuable insights for the bibliometric study, even though IDs may not fully represent the article content. The findings from both keyword types are therefore considered.

Table 7. Most Frequently Used Keywords

Keywords Plus (ID)		Author's Keywords (DE)	
Words	**f**	**Words**	**f**
students	17	chatgpt	16
educational computing	11	programming education	15
programming education	10	artificial intelligence	11
language model	9	computer science education	8
artificial intelligence	6	large language models	6
chatbots	6	generative ai	3
teaching	6	intelligent tutoring systems	3
chatgpt	5	large language model	3
computer science education	5	programming	3
engineering education	5	ai in education	2

The IDs include terms such as “students”, “educational computing”, “programming education”, “language model”, “artificial intelligence”, “chatbots”, “teaching”, “chatgpt”, “computer science education” and “engineering education”. DEs include terms like “chatgpt”, “programming education”, “artificial intelligence”, “computer science education”, “large language models”, “generative ai”, “intelligent tutoring systems”, “large language model”, “programming” and “ai in education”. Overall, the terms in IDs and DEs exhibit a number of shared concepts, including “chatgpt”, “programming education”, “artificial intelligence”, “computer science education” and “large language models”. But, the order of the terms is different between the IDs and DEs. For example, chatgpt is first order in DEs, while it is 8th order in IDs, which means that the researchers are focusing on using chatgpt for teaching computer programming. The difference between the concepts emphasized in the publication titles and the keywords may be attributed to the fact that authors often highlight different aspects in their titles compared to the key themes captured by their keywords. In addition, IDs provide more general terms such as "chatbots," "teaching," "students," "educational computing," and "language model," while DEs provide more specific terms such as "chatgpt," "generative ai," and "intelligent tutoring system". For instance, the term "chatbots" which appears in the top ten list of IDs but not in the DEs, is a more generic term than the term “chatgpt”.

RQ4. Existing Research Themes and Future Directions of generative AI in computer programming education

This section examines the relationships between terms and illuminates how generative AI is utilized within the computer programming education. This section explores insights gleaned from the co-occurrence network analysis and examines the findings of the thematic map analysis, shedding light on potential avenues for future research.

Co-Occurrence Network

The analysis of keyword co-occurrences is employed to uncover potential research areas, their interconnections, and the insights embedded within thematic clusters in this research domain [38]. The author keywords (DEs) are used as the fundamental unit of analysis to capture the primary ideas in the documents within the co-occurrence network.

The Scopus database included 177 author keywords. Vosviewer has been used with the default parameters as the full counting method has been selected and the minimum number of occurrences of a keyword has been selected as 2. Figure 2 shows the co-occurrence network of author keywords.

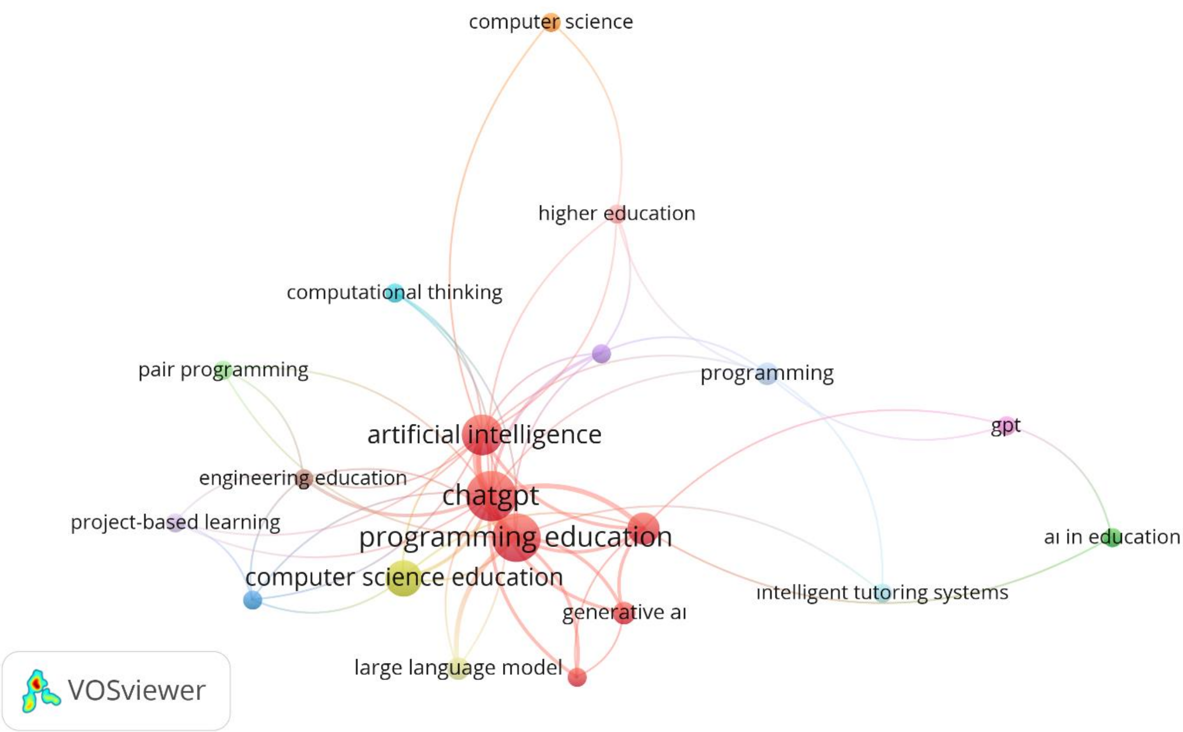

Figure 2. Author Keywords Co-occurrence Network

The central and bigger nodes in Figure 2 represent the terms with the highest occurrence frequencies within the co-occurrence network. The connections among the nodes in the co-occurrence network determine the grouping of keywords into clusters. The terms at the center of the network with the largest nodes, such as "ChatGPT", "programming education", "artificial intelligence", "copilot", and "generative AI", are the most prevalent concepts that frequently appear together in the research papers. Additionally, these terms are closely associated with concepts such as "computer science education", "large language models", "engineering education", "computational thinking", "programming", "higher education", “collaborative learning” and "project-based learning". The prominence of terms such as "ChatGPT" and "programming education" underscores a growing interest in integrating advanced AI tools into pedagogical frameworks. Moreover, some more generic keywords like "computer science", "gpt", and "ai in education" have relatively fewer co-occurrences because they are broader concepts compared to the more specific terms.

The co-occurrence of terms like "generative AI" and "ChatGPT" with pedagogical concepts such as "collaborative learning", "project-based learning", and "computational thinking" is noteworthy. The interconnectedness of these terms with concepts like "collaborative learning" and "project-based learning" indicates a potential shift towards innovative teaching methodologies that leverage AI to enhance student engagement and computational thinking. This analysis highlights the necessity for further exploration into the implications of generative AI on educational practices and curricular development.

Thematic Map

The researchers conducted thematic map analysis using author keywords to explore the prominence of various themes and identify areas warranting further research in the literature on generative AI in computer programming education. The researchers set the following parameters for their thematic map analysis: limiting the analysis to 250 words, requiring a minimum of 5 occurrences per cluster, using 3 labeled terms per cluster, and employing the "walktrap clustering algorithm" proposed by [58]. Synonymous terms, such as "generative artificial intelligence" and "generative ai," were consolidated prior to conducting the thematic map analysis. The thematic maps are defined by two primary dimensions: centrality and density. Centrality denotes the significance of a theme, while density reflects its level of development [59]. The thematic map depicted in Figure 3 presents a comprehensive overview of the research field, facilitating the identification of various themes and their relative positioning within the domain based on their significance and development.

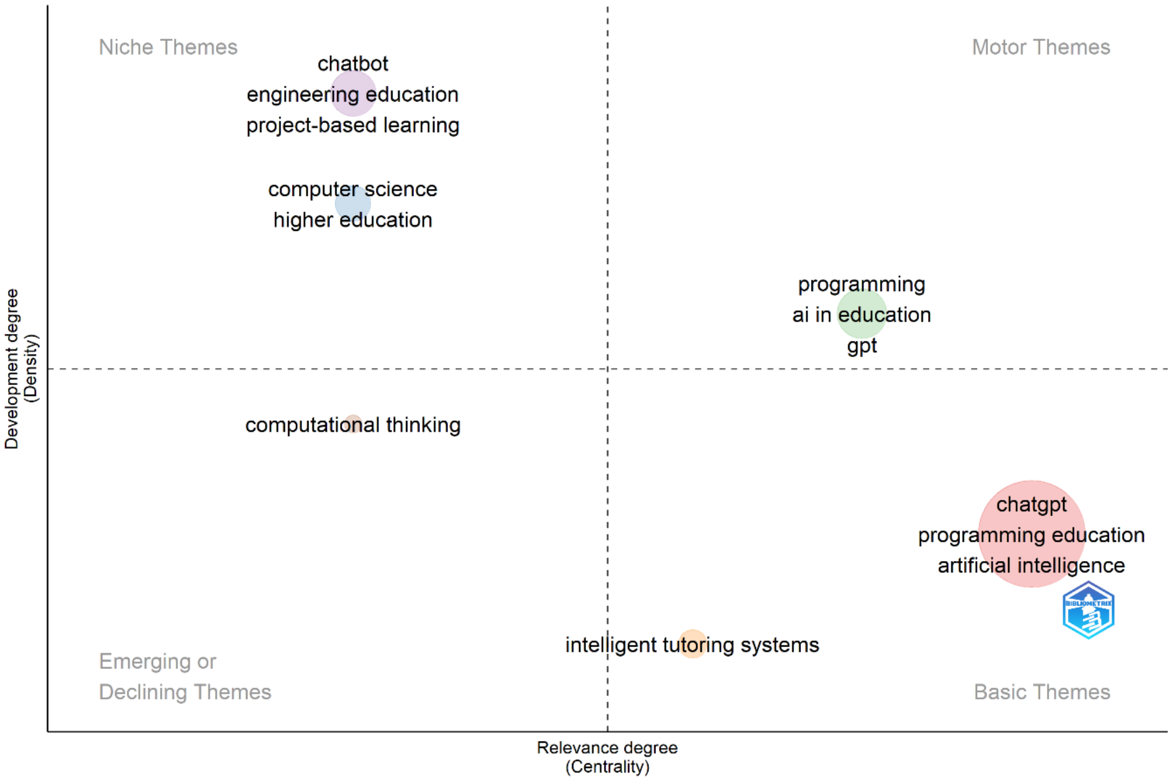

Figure 3. The Thematic Map of Author Keywords.

Figure 3 is divided into four quadrants: i) motor themes, ii) basic themes, iii) niche themes, and iv) emerging themes. Motor themes are pivotal within the research landscape, as they are both prominent and well-established. Niche themes, though well-developed, hold a relatively limited overall significance within the research landscape. Emerging themes are characterized by their isolated and underdeveloped nature, whereas basic themes are significant within the field but necessitate further exploration and maturation to reach their full potential.

This study identifies author keywords under 3 motors, 4 basics, 5 niches, and 1 emerging themes. The first research themes, "Programming," "AI in Education," and "GPT," under the motor themes category, describe the general framework of research on generative AI in computer programming education. The research themes "chatgpt", "programming education", "artificial intelligence" and "intelligent tutoring systems" which are categorized as basic themes can be viewed as core concepts in the field of generative AI in computer programming education. The niche themes identified in the analysis of the thematic map consist of 5 research themes: “chatbot”, “"engineering education”, “project-based learning”, “computer science” and “higher education”. While these niche themes have garnered growing interest from researchers, their relevance to the aforementioned research field still requires further improvement. The thematic map analysis also revealed 1 emerging theme that warrant further exploration: “computational thinking”. The "computational thinking" theme exhibits a relatively small number of associated author keywords compared to other themes, indicating that it is not yet fully developed or mature. Consequently, there is a clear need for further research that focuses on this emerging theme.

IV. CONCLUSION

The present study has explored the burgeoning field of generative AI in programming education through a comprehensive bibliometric analysis, revealing significant trends, influential research, and core themes within this dynamic area. As generative AI technologies, such as ChatGPT and GitHub Copilot, increasingly permeate educational contexts, particularly in programming instruction, it becomes imperative to understand the current research patterns and trends to shed light for future research. The findings of this study highlight the need for a careful understanding of how these tools impact educational practices.

Our findings highlight a marked increase in research output related to generative AI in programming education, particularly in the years 2023-2024, indicating a growing recognition of its relevance and potential in this field. The analysis identified 48 publications, predominantly research articles, underscoring the academic community's commitment to investigating the intersection of AI and educational practices. The bibliometric analysis also illuminated influential authors, journals, and countries contributing to this research domain. Notably, the leading journals and authors have established themselves as key contributors in the literature surrounding generative AI in programming education. The analysis of core terms and keyword co-occurrence reveals a focus on critical themes such as "programming education," "artificial intelligence," and "ChatGPT," indicating a concentrated effort to understand how these technologies can be effectively leveraged to foster learning. Furthermore, the thematic map analysis revealed key research themes, including motor themes like "Programming" and "AI in Education," and emerging themes such as "computational thinking," which warrant further exploration.

In conclusion, while generative AI holds transformative potential for programming education, it is imperative that future research continues to explore its implications, focusing on pedagogical strategies and the long-term effects on student learning outcomes. The insights gained from this bibliometric analysis serve as a foundation for guiding subsequent investigations and fostering a comprehensive understanding of the role of generative AI in shaping the future of computer programming education.

REFERENCES

[1] W. M. Lim, A. Gunasekara, J. L. Pallant, J. I. Pallant, and E. Pechenkina, "Generative AI and the future of education: Ragnarök or reformation? A paradoxical perspective from management educators," *Int. J. Manag. Educ.*, vol. 21, no. 2, p. 100790, 2023, doi: https://doi.org/10.1016/j.ijme.2023.100790.

[2] K.-B. Ooi *et al.*, "The potential of generative artificial intelligence across disciplines: Perspectives and future directions," *J. Comput. Inf. Syst.*, pp. 1–32, 2023, doi: https://doi.org/10.1080/08874417.2023.2261010.

[3] H. Polat, "Transforming education with artificial intelligence: Shaping the path forward," in *Empowering Education: Exploring the Potential of Artificial Intelligence*, 2023, pp. 3–20. [Online]. Available: www.istes.org

[4] B. A. Becker, P. Denny, J. Finnie-Ansley, A. Luxton-Reilly, J. Prather, and E. A. Santos, "Programming Is Hard – Or at Least It Used to Be: Educational Opportunities And Challenges of AI Code Generation," *Cornell University*, vol. undefined, no. undefined. Jan. 2022. doi: 10.48550/arxiv.2212.01020.

[5] P. Denny *et al.*, "Computing Education in the Era of Generative AI," *Assoc. Comput. Mach.*, vol. 67, no. 2, pp. 56–67, Jan. 2024, doi: 10.1145/3624720.

[6] C. Zastudil, M. Rogalska, C. Kapp, J. Vaughn, and S. MacNeil, "Generative ai in computing education: Perspectives of students and instructors," in *2023 IEEE Frontiers in Education Conference (FIE)*, IEEE, 2023, pp. 1–9. doi: https://doi.org/10.1109/fie58773.2023.10343467.

[7] N. A. Aarti, "Generative Ai in Software Development : an Overview and Evaluation of Modern Coding Tools," *Int. J. Multidiscip. Res.*, vol. 6, no. 3, 2024, doi: 10.36948/ijfmr.2024.v06i03.23271.

[8] J. T. Liang, C. Yang, and B. A. Myers, "A Large-Scale Survey on the Usability of AI Programming Assistants: Successes and Challenges," 2024, doi: 10.1145/3597503.3608128.

[9] S. Abdulla, S. S. Ismail, Y. M. Fawzy, and A. Elhag, "Using ChatGPT in Teaching Computer Programming and Studying its Impact on Students Performance," *Electron. J. E-Learn.*, vol. 22, no. 6, pp. 66–81, 2024, doi: 10.34190/ejel.22.6.3380.

[10] M. Amoozadeh *et al.*, "Student-AI Interaction: A Case Study of CS1 students," in *Proceedings of the 24th Koli Calling International Conference on Computing Education Research*, 2024, pp. 1–13.

[11] Z. Epstein *et al.*, "Art and the science of generative AI," *Science*, vol. 380, no. 6650, pp. 1110–1111, 2023.

[12] K. Kadaruddin, “Empowering Education through Generative AI: Innovative Instructional Strategies for Tomorrow’s Learners,” *Int. J. Bus. Law Educ.*, 2023, doi: 10.56442/ijble.v4i2.215.

[13] S. Altares-López, J. M. Bengochea-Guevara, C. Ranz, H. Montes, and Á. Ribeiro, “Generative AI: The power of the new education,” *Cornell University*, vol. undefined, no. undefined. May 2024. doi: 10.48550/arxiv.2405.13487.

[14] S. Boguslawski, R. Deer, and M. G. Dawson, “Programming education and learner motivation in the age of generative AI: student and educator perspectives,” *Information and Learning Science*. Emerald Publishing, 2024. doi: 10.1108/ILS-10-2023-0163.

[15] E. Dickey, A. Bejarano, and C. Garg, “Innovating Computer Programming Pedagogy: The AI-Lab Framework for Generative AI Adoption,” *ArXiv Prepr. ArXiv230812258*, 2023, doi: https://doi.org/10.48550/arxiv.2308.12258.

[16] K. Łodzikowski, P. W. Foltz, and J. T. Behrens, “Generative AI and Its Educational Implications,” *Springer International Publishing*, vol. undefined, no. undefined. pp. 35–57, Jan. 2024. doi: 10.1007/978-3-031-64487-0_2.

[17] R. F. Mello, E. Freitas, F. D. Pereira, L. Cabral, P. Tedesco, and G. Ramalho, “Education in the age of Generative AI: Context and Recent Developments,” *ArXiv Prepr. ArXiv230912332*, 2023.

[18] S. Lau and P. Guo, “From" Ban it till we understand it" to" Resistance is futile": How university programming instructors plan to adapt as more students use AI code generation and explanation tools such as ChatGPT and GitHub Copilot,” in *Proceedings of the 2023 ACM Conference on International Computing Education Research-Volume 1*, 2023, pp. 106–121.

[19] A. H. Mohammadkhani, C. Tantithamthavorn, and H. Hemmati, “Explainable AI for Pre-Trained Code Models: What Do They Learn? When They Do Not Work?,” *arXiv.org*, vol. abs/2211.12821, 2022, doi: 10.48550/arXiv.2211.12821.

[20] M. Kazemitabaar, J. Chow, C. K. T. Ma, B. J. Ericson, D. Weintrop, and T. Grossman, “Studying the effect of AI code generators on supporting novice learners in introductory programming,” in *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, 2023, pp. 1–23.

[21] B. R. Mohan, A. Chainani, A. K. Kartheek, C. Praneeth, and K. Shakelli, “IntelliDev: AI-Powered Command Line Tool for Developers,” *Int. J. Eng. Technol. Manag. Sci.*, vol. 8, no. 3, pp. 161–165, 2024, doi: 10.46647/ijetms.2024.v08i03.019.

[22] R. A. Poldrack, T. Lu, and G. Beguš, “AI-assisted coding: Experiments with GPT-4,” *arXiv.org*, vol. abs/2304.13187, 2023, doi: 10.48550/arXiv.2304.13187.

[23] R. Zviel-Girshin, “The Good and Bad of AI Tools in Novice Programming Education,” *Educ. Sci.*, vol. 14, no. 10, p. 1089, 2024, doi: https://doi.org/10.3390/educsci14101089.

[24] C.-K. Chang and L.-C.-T. Chien, “Enhancing Academic Performance with Generative AI-Based Quiz Platform,” pp. 193–195, 2024, doi: 10.1109/icalt61570.2024.00062.

[25] M. Lehmann, P. B. Cornelius, and F. J. Sting, “AI Meets the Classroom: When Does ChatGPT Harm Learning?,” 2024, doi: 10.48550/arxiv.2409.09047.

[26] D. M. Johnson, W. Doss, and C. M. Estepp, “Agriculture students’ use of generative artificial intelligence for microcontroller programming,” *Nat. Sci. Educ.*, vol. 53, no. 2, 2024, doi: 10.1002/nse2.20155.

[27] J. Prather *et al.*, “The Widening Gap: The Benefits and Harms of Generative AI for Novice Programmers,” in *Proceedings of the 2024 ACM Conference on International Computing Education Research-Volume 1*, 2024, pp. 469–486.

[28] Y. Xue, H. Chen, G. R. Bai, R. Tairas, and Y. Huang, “Does ChatGPT Help With Introductory Programming? An Experiment of Students Using ChatGPT in CS1,” in *Proceedings of the 46th International Conference on Software Engineering: Software Engineering Education and Training*, 2024, pp. 331–341.

[29] J. O. Wecks, J. Voshaar, B. J. Plate, and J. Zimmermann, “Generative AI Usage and Academic Performance,” *ArXiv Prepr. ArXiv240419699*, 2024, doi: https://doi.org/10.48550/arxiv.2404.19699.

[30] J. Liu and S. Li, “Toward Artificial Intelligence-Human Paired Programming: A Review of the Educational Applications and Research on Artificial Intelligence Code-Generation Tools,” *J. Educ. Comput. Res.*, 2024, doi: 10.1177/07356331241240460.

[31] P. Haindl and G. Weinberger, “Students’ Experiences of Using ChatGPT in an Undergraduate Programming Course,” *IEEE Access*, vol. 12, pp. 43519–43529, 2024, doi: https://doi.org/10.1109/access.2024.3380909.

[32] W. E. McGrath, “The unit of analysis (objects of study) in bibliometrics and scientometrics,” *Springer Nat. Neth.*, vol. 35, no. 2, pp. 257–264, Feb. 1996, doi: 10.1007/bf02018483.

[33] J. Vlachý, “Scientometrics — What to do?,” *Springer Nat. Neth.*, vol. 30, no. 2, pp. 521–527, Jun. 1994, doi: 10.1007/bf02018134.

[34] R. Rousseau and X. Hu, “Under-cited influential work by Eugene Garfield,” *Springer Nat. Neth.*, vol. 114, no. 2, pp. 651–657, Sep. 2017, doi: 10.1007/s11192-017-2530-8.

[35] W. W. Hood and C. S. Wilson, “The Literature of Bibliometrics, Scientometrics, and Informetrics,” *Scientometrics*, vol. 52, no. 2, pp. 291–314, Oct. 2001, doi: 10.1023/A:1017919924342.

[36] N. Donthu, S. Kumar, D. Mukherjee, N. Pandey, and W. M. Lim, “How to conduct a bibliometric analysis: An overview and guidelines,” *J. Bus. Res.*, vol. 133, pp. 285–296, Sep. 2021, doi: 10.1016/j.jbusres.2021.04.070.

[37] R. Heersmink, J. van den Hoven, N. J. van Eck, and J. van den Berg, “Bibliometric mapping of computer and information ethics,” *Ethics Inf. Technol.*, vol. 13, no. 3, pp. 241–249, Sep. 2011, doi: 10.1007/s10676-011-9273-7.

[38] H. Polat, A. Topuz, M. Yıldız, E. Taşlıbeyaz, and E. Kurşun, "A bibliometric analysis of research on ChatGPT in education," *Int. J. Technol. Educ. IJTE*, vol. 7, no. 1, 2024, doi: https://doi.org/10.46328/ijte.606.
[39] M. Pradana, H. P. Elisa, and S. Syarifuddin, "Discussing ChatGPT in education: A literature review and bibliometric analysis," *Taylor & Francis*, vol. 10, no. 2. Aug. 2023. doi: 10.1080/2331186x.2023.2243134.
[40] P. Song and X. Wang, "A bibliometric analysis of worldwide educational artificial intelligence research development in recent twenty years," *Springer Sci. Media*, vol. 21, no. 3, pp. 473–486, Aug. 2020, doi: 10.1007/s12564-020-09640-2.
[41] M. Aria and C. Cuccurullo, "bibliometrix: An R-tool for comprehensive science mapping analysis," *J. Informetr.*, vol. 11, no. 4, pp. 959–975, 2017, doi: https://doi.org/10.1016/j.joi.2017.08.007.
[42] N. J. van Eck and L. Waltman, "Software survey: VOSviewer, a computer program for bibliometric mapping," *Scientometrics*, vol. 84, no. 2, pp. 523–538, Aug. 2010, doi: 10.1007/s11192-009-0146-3.
[43] V. K. Singh, P. Singh, M. Karmakar, J. Leta, and P. Mayr, "The journal coverage of Web of Science, Scopus and Dimensions: A comparative analysis," *Springer Nat. Neth.*, vol. 126, no. 6, pp. 5113–5142, Mar. 2021, doi: 10.1007/s11192-021-03948-5.
[44] J. Zhu and W. Liu, "A tale of two databases: the use of Web of Science and Scopus in academic papers," *Springer Nat. Neth.*, vol. 123, no. 1, pp. 321–335, Feb. 2020, doi: 10.1007/s11192-020-03387-8.
[45] Abad-Segura, F. J. Cortés-García, and L. J. B. Ureña, "The Sustainable Approach to Corporate Social Responsibility: A Global Analysis and Future Trends," *Multidiscip. Digit. Publ. Inst.*, vol. 11, no. 19, pp. 5382–5382, Sep. 2019, doi: 10.3390/su11195382.
[46] W. M. Sweileh, "Research trends on human trafficking: a bibliometric analysis using Scopus database," *BioMed Cent.*, vol. 14, no. 1, Nov. 2018, doi: 10.1186/s12992-018-0427-9.
[47] Md. M. Rahman and Y. Watanobe, "ChatGPT for Education and Research: Opportunities, Threats, and Strategies," *Applied Sciences (Switzerland)*, vol. 13, no. 9. MDPI, 2023. doi: 10.3390/app13095783.
[48] R. Yilmaz and F. G. Karaoglan Yilmaz, "The effect of generative artificial intelligence (AI)-based tool use on students' computational thinking skills, programming self-efficacy and motivation," *Computers and Education: Artificial Intelligence*, vol. 4. Elsevier B.V., 2023. doi: 10.1016/j.caeai.2023.100147.
[49] C. W. Okonkwo and A. Ade-Ibijola, "Python-bot: A chatbot for teaching python programming," *Engineering Letters*, vol. 29, no. 1. International Association of Engineers, pp. 25–34, 2021.
[50] Md. M. Rahman, Y. Watanobe, and K. Nakamura, "Source Code assessment and classification based on estimated error probability using attentive lstm language model and its application in programming education," *Applied Sciences (Switzerland)*, vol. 10, no. 8. MDPI AG, 2020. doi: 10.3390/APP10082973.
[51] N.-T. Le, S. Strickroth, S. Gross, and N. Pinkwart, "A review of AI-supported tutoring approaches for learning programming," *Studies in Computational Intelligence*, vol. 479. Springer Verlag, pp. 267–279, 2013. doi: 10.1007/978-3-319-00293-4_20.
[52] K.-W. Han, E. Lee, and Y. Lee, "The impact of a peer-learning agent based on pair programming in a programming course," *IEEE Transactions on Education*, vol. 53, no. 2. pp. 318–327, 2010. doi: 10.1109/TE.2009.2019121.
[53] J. McBroom, I. Koprinska, and K. Yacef, "A Survey of Automated Programming Hint Generation: The HINTS Framework," *ACM Computing Surveys*, vol. 54, no. 8. Association for Computing Machinery, 2022. doi: 10.1145/3469885.
[54] C. A. G. da Silva, F. N. Ramos, R. V. de Moraes, and E. L. dos Santos, "ChatGPT: Challenges and Benefits in Software Programming for Higher Education," *Sustainability (Switzerland)*, vol. 16, no. 3. Multidisciplinary Digital Publishing Institute (MDPI), 2024. doi: 10.3390/su16031245.
[55] L. Jacques, "Teaching CS-101 at the Dawn of ChatGPT," *ACM Inroads*, vol. 14, no. 2. Association for Computing Machinery, pp. 40–46, 2023. doi: 10.1145/3595634.
[56] D. Sun, A. Boudouaia, C. Zhu, and Y. Li, "Would ChatGPT-facilitated programming mode impact college students' programming behaviors, performances, and perceptions? An empirical study," *International Journal of Educational Technology in Higher Education*, vol. 21, no. 1. Springer Science and Business Media Deutschland GmbH, 2024. doi: 10.1186/s41239-024-00446-5.
[57] J. Zhang, Y. Qi, F. Zheng, C. Long, Z. Lu, and Z. Duan, "Comparing keywords plus of WOS and author keywords: A case study of patient adherence research," *Wiley*, vol. 67, no. 4, pp. 967–972, Jan. 2015, doi: 10.1002/asi.23437.
[58] A. Lancichinetti and S. Fortunato, "Community detection algorithms: A comparative analysis," *Am. Phys. Soc.*, vol. 80, no. 5, Nov. 2009, doi: 10.1103/physreve.80.056117.
[59] M. J. Cobo, A. G. López-Herrera, E. Herrera-Viedma, and F. Herrera, "An approach for detecting, quantifying, and visualizing the evolution of a research field: A practical application to the Fuzzy Sets Theory field," *Elsevier BV*, vol. 5, no. 1, pp. 146–166, Nov. 2010, doi: 10.1016/j.joi.2010.10.002.

Appendix 1

Num.	Type	Year	author(s)	Title	doi
1	ARTICLE	2024	OLARI V;ROMEIKE R	DATA-RELATED CONCEPTS FOR ARTIFICIAL INTELLIGENCE EDUCATION IN K-12	10.1016/j.caeo.2024.100196
2	ARTICLE	2024	ZVIEL-GIRSHIN R	THE GOOD AND BAD OF AI TOOLS IN NOVICE PROGRAMMING EDUCATION	10.3390/educsci14101089
3	ARTICLE	2024	AKÇAPINAR G;SIDAN E	AI CHATBOTS IN PROGRAMMING EDUCATION: GUIDING SUCCESS OR ENCOURAGING PLAGIARISM	10.1007/s44163-024-00203-7
4	ARTICLE	2024	SILVA CAGD;RAMOS FN;DE MORAES RV;SANTOS ELD	CHATGPT: CHALLENGES AND BENEFITS IN SOFTWARE PROGRAMMING FOR HIGHER EDUCATION	10.3390/su16031245
5	ARTICLE	2024	FRUETT F;PEREIRA BARBOSA F;CARDOSO ZAMPOLLI FRAGA S;IVO ARAGAO GUIMARAES P	EMPOWERING STEAM ACTIVITIES WITH ARTIFICIAL INTELLIGENCE AND OPEN HARDWARE: THE BITDOGLAB	10.1109/TE.2024.3377555
6	ARTICLE	2024	GROOTHUIJSEN S;VAN DEN BEEMT A;REMMERS JC;VAN MEEUWEN LW	AI CHATBOTS IN PROGRAMMING EDUCATION: STUDENTS’ USE IN A SCIENTIFIC COMPUTING COURSE AND CONSEQUENCES FOR LEARNING	10.1016/j.caeai.2024.100290
7	ARTICLE	2024	ELLIS ME;CASEY KM;HILL G	CHATGPT AND PYTHON PROGRAMMING HOMEWORK	10.1111/dsji.12306
8	ARTICLE	2024	GELLER J	TEACHING COMPUTING AND EXPERIENCING GRIEF IN THE YEAR 1 AC (AFTER CHATGPT)	10.3389/feduc.2024.1364255
9	ARTICLE	2024	TLILI A	CAN ARTIFICIAL INTELLIGENCE (AI) HELP IN COMPUTER SCIENCE EDUCATION? A META-ANALYSIS APPROACH; [¿PUEDE AYUDAR LA INTELIGENCIA ARTIFICIAL (IA) EN LA EDUCACIÓN EN CIENCIAS DE LA COMPUTACIÓN? UN ENFOQUE METAANALÍTICO]	10.22550/2174-0909.4172
10	ARTICLE	2024	BOGUSLAWSKI S;DEER R;DAWSON MG	PROGRAMMING EDUCATION AND LEARNER MOTIVATION IN THE AGE OF GENERATIVE AI: STUDENT AND EDUCATOR PERSPECTIVES	10.1108/ILS-10-2023-0163
11	ARTICLE	2024	OUYANG F;GUO M;ZHANG N;BAI X;JIAO P	COMPARING THE EFFECTS OF INSTRUCTOR MANUAL FEEDBACK AND CHATGPT INTELLIGENT FEEDBACK ON COLLABORATIVE PROGRAMMING IN CHINA'S HIGHER EDUCATION	10.1109/TLT.2024.3486749
12	ARTICLE	2024	BOUDIA C;KRISMADINATA	EARLY INSIGHTS INTO SLA WITH CHATGPT: NAVIGATING CS TEACHERS AND STUDENT PERSPECTIVES IN AN OPINION-BASED EXPLORATION	10.55214/25768484.v8i5.1729
13	ARTICLE	2024	LIAO J;ZHONG L;ZHE L;XU H;LIU M;XIE T	SCAFFOLDING COMPUTATIONAL THINKING WITH CHATGPT	10.1109/TLT.2024.3392896
14	ARTICLE	2024	ROCHA A;SOUSA L;ALVES M;SOUSA A	THE UNDERLYING POTENTIAL OF NLP FOR MICROCONTROLLER PROGRAMMING EDUCATION	10.1002/cae.22778

15	ARTICLE	2024	ZHANG Z;WEN L;JIANG Y;LIU Y	EVALUATE CHAT-GPT'S PROGRAMMING CAPABILITY IN SWIFT THROUGH REAL UNIVERSITY EXAM QUESTIONS	10.1002/spe.3330
16	ARTICLE	2024	MONTELLA R;DE VITA CG;MELLONE G;CIRICILLO T;CARAMIELLO D;DI LUCCIO D;KOSTA S;DAMAŠEVIČIUS R;MASKELIŪNAS R;QUEIRÓS R;SWACHA J	LEVERAGING LARGE LANGUAGE MODELS TO SUPPORT AUTHORING GAMIFIED PROGRAMMING EXERCISES †	10.3390/app14188344
17	ARTICLE	2024	MENDONÇA NC	EVALUATING CHATGPT-4 VISION ON BRAZIL'S NATIONAL UNDERGRADUATE COMPUTER SCIENCE EXAM	10.1145/3674149
18	ARTICLE	2024	ABDULSAHIB AK;MOHAMMED R;AHMED AL;JABER MM	ARTIFICIAL INTELLIGENCE BASED COMPUTER VISION ANALYSIS FOR SMART EDUCATION INTERACTIVE VISUALIZATION	10.54216/FPA.150221
19	ARTICLE	2024	HAINDL P;WEINBERGER G	DOES CHATGPT HELP NOVICE PROGRAMMERS WRITE BETTER CODE? RESULTS FROM STATIC CODE ANALYSIS	10.1109/ACCESS.2024.3445432
20	ARTICLE	2024	JOŠT G;TANESKI V;KARAKATIČ S	THE IMPACT OF LARGE LANGUAGE MODELS ON PROGRAMMING EDUCATION AND STUDENT LEARNING OUTCOMES	10.3390/app14104115
21	ARTICLE	2024	SUN D;BOUDOUAIA A;ZHU C;LI Y	WOULD CHATGPT-FACILITATED PROGRAMMING MODE IMPACT COLLEGE STUDENTS' PROGRAMMING BEHAVIORS, PERFORMANCES, AND PERCEPTIONS? AN EMPIRICAL STUDY	10.1186/s41239-024-00446-5
22	ARTICLE	2024	ZHANG J;CAMBRONERO JP;GULWANI S;LE V;PISKAC R;SOARES G;VERBRUGGEN G	PYDEX: REPAIRING BUGS IN INTRODUCTORY PYTHON ASSIGNMENTS USING LLMS	10.1145/3649850
23	ARTICLE	2024	JING Y;WANG H;CHEN X;WANG C	WHAT FACTORS WILL AFFECT THE EFFECTIVENESS OF USING CHATGPT TO SOLVE PROGRAMMING PROBLEMS? A QUASI-EXPERIMENTAL STUDY	10.1057/s41599-024-02751-w
24	ARTICLE	2024	HAINDL P;WEINBERGER G	STUDENTS' EXPERIENCES OF USING CHATGPT IN AN UNDERGRADUATE PROGRAMMING COURSE	10.1109/ACCESS.2024.3380909
25	ARTICLE	2024	LEE S;SONG K-S	TEACHERS' AND STUDENTS' PERCEPTIONS OF AI-GENERATED CONCEPT EXPLANATIONS: IMPLICATIONS FOR INTEGRATING GENERATIVE AI IN COMPUTER SCIENCE EDUCATION	10.1016/j.caeai.2024.100283
26	ARTICLE	2024	WAN H;LUO H;LI M;LUO X	AUTOMATED PROGRAM REPAIR FOR INTRODUCTORY PROGRAMMING ASSIGNMENTS	10.1109/TLT.2024.3403710
27	ARTICLE	2024	HUMBLE N;BOUSTEDT J;HOLMGREN H;MILUTINOVIC G;SEIPEL S;ÖSTBERG A-S	CHEATERS OR AI-ENHANCED LEARNERS: CONSEQUENCES OF CHATGPT FOR PROGRAMMING EDUCATION	10.34190/ejel.21.5.3154

28	ARTICLE	2024	ZHONG H-X;CHANG J-H;LAI C-F;CHEN P-W;KU S-H;CHEN S-Y	INFORMATION UNDERGRADUATE AND NON-INFORMATION UNDERGRADUATE ON AN ARTIFICIAL INTELLIGENCE LEARNING PLATFORM: AN ARTIFICIAL INTELLIGENCE ASSESSMENT MODEL USING PLS-SEM ANALYSIS	10.1007/s10639-023-11961-9
29	ARTICLE	2024	LIU J;LI S	TOWARD ARTIFICIAL INTELLIGENCE-HUMAN PAIRED PROGRAMMING: A REVIEW OF THE EDUCATIONAL APPLICATIONS AND RESEARCH ON ARTIFICIAL INTELLIGENCE CODE-GENERATION TOOLS	10.1177/07356331241240460
30	ARTICLE	2024	HUSAIN AJA	POTENTIALS OF CHATGPT IN COMPUTER PROGRAMMING: INSIGHTS FROM PROGRAMMING INSTRUCTORS	10.28945/5240
31	ARTICLE	2024	KOSAR T;OSTOJIĆ D;LIU YD;MERNIK M	COMPUTER SCIENCE EDUCATION IN CHATGPT ERA: EXPERIENCES FROM AN EXPERIMENT IN A PROGRAMMING COURSE FOR NOVICE PROGRAMMERS	10.3390/math12050629
32	ARTICLE	2024	ESSEL HB;VLACHOPOULOS D;NUNOO-MENSAH H;AMANKWA JO	EXPLORING THE IMPACT OF VOICEBOTS ON MULTIMEDIA PROGRAMMING EDUCATION AMONG GHANAIAN UNIVERSITY STUDENTS	10.1111/bjet.13504
33	ARTICLE	2024	YANG ACM;LIN J-Y;LIN C-Y;OGATA H	ENHANCING PYTHON LEARNING WITH PYTUTOR: EFFICACY OF A CHATGPT-BASED INTELLIGENT TUTORING SYSTEM IN PROGRAMMING EDUCATION	10.1016/j.caeai.2024.100309
34	ARTICLE	2024	ROLDAN-ALVAREZ D;MESA FJ	INTELLIGENT DEEP-LEARNING TUTORING SYSTEM TO ASSIST INSTRUCTORS IN PROGRAMMING COURSES	10.1109/TE.2023.3331055
35	ARTICLE	2023	PHILBIN CA	EXPLORING THE POTENTIAL OF ARTIFICIAL INTELLIGENCE PROGRAM GENERATORS IN COMPUTER PROGRAMMING EDUCATION FOR STUDENTS	10.1145/3610406
36	ARTICLE	2023	DENGEL A;GEHRLEIN R;FERNES D;GÖRLICH S;MAURER J;PHAM HH;GROßMANN G;EISERMANN NDG	QUALITATIVE RESEARCH METHODS FOR LARGE LANGUAGE MODELS: CONDUCTING SEMI-STRUCTURED INTERVIEWS WITH CHATGPT AND BARD ON COMPUTER SCIENCE EDUCATION	10.3390/informatics10040078
37	ARTICLE	2023	RAHMAN MM;WATANOBE Y	CHATGPT FOR EDUCATION AND RESEARCH: OPPORTUNITIES, THREATS, AND STRATEGIES	10.3390/app13095783
38	ARTICLE	2023	AZAIZ I;DECKARM O;STRICKROTH S	AI-ENHANCED AUTO-CORRECTION OF PROGRAMMING EXERCISES: HOW EFFECTIVE IS GPT-3.5?	10.3991/ijep.v13i8.45621
39	ARTICLE	2023	YILMAZ R;KARAOGLAN YILMAZ FG	THE EFFECT OF GENERATIVE ARTIFICIAL INTELLIGENCE (AI)-BASED TOOL USE ON STUDENTS' COMPUTATIONAL THINKING SKILLS, PROGRAMMING SELF-	10.1016/j.caeai.2023.100147

				EFFICACY AND MOTIVATION	
40	ARTICLE	2023	JACQUES L	TEACHING CS-101 AT THE DAWN OF CHATGPT	10.1145/3595634
41	ARTICLE	2023	CHUNG C-Y;HSIAO I-H;LIN Y-L	AI-ASSISTED PROGRAMMING QUESTION GENERATION: CONSTRUCTING SEMANTIC NETWORKS OF PROGRAMMING KNOWLEDGE BY LOCAL KNOWLEDGE GRAPH AND ABSTRACT SYNTAX TREE	10.1080/15391523.2022.2123872
42	ARTICLE	2023	KWAK M;JENKINS J;KIM J	ADAPTIVE PROGRAMMING LANGUAGE LEARNING SYSTEM BASED ON GENERATIVE AI	10.48009/3_iis_2023_119
43	REVIEW	2022	MCBROOM J;KOPRINSKA I;YACEF K	A SURVEY OF AUTOMATED PROGRAMMING HINT GENERATION: THE HINTS FRAMEWORK	10.1145/3469885
44	ARTICLE	2021	OKONKWO CW;ADE-IBIJOLA A	PYTHON-BOT: A CHATBOT FOR TEACHING PYTHON PROGRAMMING	
45	ARTICLE	2020	RAHMAN MM;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	10.3390/APP10082973
46	REVIEW	2013	LE N-T;STRICKROTH S;GROSS S;PINKWART N	A REVIEW OF AI-SUPPORTED TUTORING APPROACHES FOR LEARNING PROGRAMMING	10.1007/978-3-319-00293-4_20
47	ARTICLE	2010	HAN K-W;LEE E;LEE Y	THE IMPACT OF A PEER-LEARNING AGENT BASED ON PAIR PROGRAMMING IN A PROGRAMMING COURSE	10.1109/TE.2009.2019121
48	ARTICLE	1994	UENO HARUKI	INTEGRATED INTELLIGENT PROGRAMMING ENVIRONMENT FOR LEARNING PROGRAMMING	10.1016/j.caeo.2024.100196