8-2023

AI for connectivism learning: Undergraduate students' experiences of ChatGPT in advanced programming courses

Swapna GOTTIPATI
Singapore Management University, SWAPNAG@smu.edu.sg

Kyong Jin SHIM
Singapore Management University, kjshim@smu.edu.sg

Venky SHANKARARAMAN
Singapore Management University, venks@smu.edu.sg

Citation

GOTTIPATI, Swapna; SHIM, Kyong Jin; and SHANKARARAMAN, Venky. AI for connectivism learning: Undergraduate students' experiences of ChatGPT in advanced programming courses. (2023). *Proceedings of Americas Conference on Information Systems 2023: Panama, August 10-12*. 16, 1-10.
Available at: https://ink.library.smu.edu.sg/sis_research/8496

AI for Connectivism Learning - Undergraduate Students' Experiences of ChatGPT in Advanced Programming Courses

Completed Research Full Paper

Swapna Gottipati
Singapore Management University
swapnag@smu.edu.sg

Kyong Jin Shim
Singapore Management University
kjshim@smu.edu.sg

Venky Shankararaman
Singapore Management University
venks@smu.edu.sg

Abstract

Advanced programming skills are required for computing courses on merging topics, and students often struggle to develop these skills to solve complex problems. To address this challenge, faculty members provide additional lectures, practice sessions, and educational technology tools. This paper discusses the challenges faced by computer science students in developing advanced programming skills and explores the use of AI chatbots, specifically ChatGPT, as a support tool for learning. We study the engagement and effectiveness of ChatGPT in helping students learn advanced programming skills using two engagement learning frameworks (CIE and MELT) for evaluation. The study involves designing a computing lab exercise (design and code questions) for students to complete using ChatGPT and collecting data through surveys. The findings provide initial evidence that ChatGPT can be an effective tool for supporting student learning in advanced programming courses.

Keywords

Advanced programming skills, Connectivism learning, Problem solving skills, ChatGPT, Engagement Learning Models.

Introduction

Several universities develop their Computer Science (CS) curriculum around the guidelines and requirements based on ACM/IEEE Joint Curriculum recommendations. Based on the recent recommendations (Clear, 2019), emerging topics (specialized sub-areas or track of CS programs) such as data science, AI, cybersecurity, cloud computing, etc., are increasingly becoming part of the program, and several advanced programming courses based on these topics are introduced to CS undergraduates in year 3 and year 4. Such courses require advanced programming skills and students face challenges in developing such advanced programming skills for solving complex problems.

Many faculty address this challenge through additional lectures, extra practices, or more holistic approaches to support student learning. Some faculty use technology for effective learning. For example, by integrating secure coding education into the Integrated Development Environment (IDE) tool to provide a learning opportunity in the context of writing code (Whitney et al., 2015) or mobile apps (Riley, 2012), and extreme apprenticeship (Vihavaine et al., 2011), etc. Most of the above techniques are used in advanced computing courses related to topics in web security, complex Java applications, Unix/kernel programming, and multithreading. These techniques have proven to be very successful. However, maintaining and developing the tools or securing and managing apprenticeships can be a tedious process.

The theory of connectivism was introduced as a learning model suitable for the digital era, and the advancements in Artificial Intelligence (AI), provide evidence to support the ideas about knowledge connectivity (Utecht & Keller, 2019). Connectivism learning typically takes place in digital environments such as online courses, social media platforms, or online communities of practice. Learners are encouraged to use digital tools and technologies to access information, collaborate with peers, and create and share their knowledge (Dishman, 2022) and hence develop problem solving skills. Customised AI tools aid in teaching, learning, and assessment management (Chassignol et al, 2018). AI driven chatbots are also used in programming courses by several instructors (Verleger, 2018; Okonkwo, 2020). ChatGPT (https://chat.openai.com/chat) is the most recent and popular AI chatbot that covers various domains such as healthcare, legal, technology, etc. (Antaki et al., 2023; Choi et al., 2023). In education, ChatGPT can be used to assist and support learners, educators, and researchers in various ways. For example, it can provide access to a vast amount of knowledge and information from various disciplines, answer questions, explain complex concepts, as well as suggest learning resources (Zhai, 2022).

In this work, we study students' experiences of using ChatGPT as a support tool in helping them learn advanced programming skills. We study the interactions between the students and ChatGPT to understand the level of engagement and effectiveness of ChatGPT as a support tool for learning advanced programming. To evaluate the student-ChatGPT engagements we use the Connectivism learning model and apply two frameworks. Firstly, a framework for Interaction and Cognitive Engagement in Connectivist Learning (CIE) was developed by Wang, et al. (Wang et al., 2014). Secondly, Models of Engaged Learning and Teaching (MELT) framework was developed by Willison, 2018. Since several models are proposed under MELT framework, for this research we use CIE and MELT-OPS framework to study the student-ChatGPT interactions.

In our approach, we design a lab exercise that includes advanced programming questions related to the content taught. Students are expected to complete the questions in the classroom using the support of ChatGPT and submit the lab exercise to the instructor. Subsequently, they are required complete a survey questionnaire. This questionnaire is designed based on the two frameworks namely CIE and MELT-OPS to help understand the interactions and measure connectivism learning, problem solving skills as well as the usefulness of ChatGPT as a support tool for advanced programming skills development.

In this paper, we describe the engagement learning frameworks and their connection with ChatGPT as an interacting digital tool in advanced programming courses. We aim to answer two research questions; (1) What are the experiences of students using ChatGPT as a support tool for learning advanced programming skills? and (2) how effective and engaging is ChatGPT in facilitating the learning process? We present the education theories, methodology, data collection and data analysis in the following sections. We finally present findings of tool effectiveness, student sentiments and limitations of our work.

Engagement Learning Frameworks

Connectivism learning is a modern approach to learning that emphasizes the role of connections and networks in the process of acquiring knowledge and skills. This approach is based on the idea that knowledge is distributed across a network of people, technologies, and information sources, rather than being concentrated in a central authority or expert (Duke et al., 2013). In connectivism learning, learners are encouraged to make connections between different sources of information (wikis, search engines, AI tools, etc.,) and to actively participate in creating and sharing knowledge within a learning community. Learners are also expected to be self-directed and to take responsibility for their learning, rather than relying solely on a teacher or instructor (Siemens, 2005; Siemens, 2012).

Measuring networks such as teacher-learner interactions in a classroom can be challenging, as it involves capturing both verbal and non-verbal communication, as well as assessing the quality of these interactions (Dunlap et al., 2007; Englehart, 2009). One of the most common methods of measuring teacher-learner interactions is through classroom observations. Surveys and questionnaires can be used to measure students' perceptions of teacher-learner interactions. Questions can be asked about the quality of the communication, the effectiveness of the teaching methods, the level of engagement, and the level of support provided. Classroom artifacts, such as assignments can be used to measure the level of engagement and the quality of teacher-learner interactions. In our study, we focus on measuring the interactions between ChatGPT and the students using the lab exercise and survey. Recall from Section 1 that our survey model is derived based on two engagement frameworks namely CIE and MELT-OPS.

Connectivist Interaction Engagement (CIE) Framework

Wang et al. (2014), developed the CIE framework by combining the connectivism learning theory (Siemens, 2005) and Bloom's revised taxonomy (Anderson & Krathwohl, 2021). The participants have increased choice and opportunity to interact with peers according to their network literacy (Belshaw, 2013; Chung et al., 2017), the networks they belong to, and the sets they curate and with which they interact. Wang divided the interactions into four levels and explained the connections with Bloom's taxonomy across these levels (Wang et al., 2014) as below.

Operations (Laurillards, 1999): This level aims at building the interaction space such as social media (e.g., discussion forums) or personalised learning environments (e.g., coursera, ChatGPT). It is a process where learners connect with different technologies through learner-interface interaction to support their further learning. During operation interaction, the learners merely practice and remember how to operate various media and tools to build their own learning spaces.

Wayfinding (Siemens, 2012,): This level aims at building the pipeline of knowledge flow where learners are involved actively in creating and participating in groups and networks as well as taking advantage of recommendations from the operations level (e.g., interaction with ChatGPT with programming questions and explanations). In wayfinding interaction, learners must master the ways to navigate in a complex information environment and connect with different human and non-human resources. Hence they must reach higher levels of understanding (apply and evaluate the information and connection) in this process.

Sensemaking (Siemens, 2012): This level involves connection building and idea formation and includes stages such as information aggregation/sharing, discussion/negotiation, reflection, and decision making (e.g., critique and analyse answers from ChatGPT). Sensemaking is a pattern recognition process, so the top five categories of Bloom's revised taxonomy are each involved in it, especially applying, analysing, and evaluating. It is also the foundation for the innovation interaction level.

Innovation: This level includes the knowledge growth process by further reflection and presentation of sensemaking results. It is the most challenging and critical interaction for learners. Innovation interaction focuses on the expression of ideas, models, or theory by artifact creation and innovation to enhance and build new social, technological, and informational connections (e.g., compare and improve the ChatGPT answers with more ideas or external knowledge). It thus engages learners at the creation level of Bloom's revised taxonomy.

To analyse the role of ChatGPT in advanced programming development skills, we study the students' interactions with ChatGPT using the CIE framework. The types of interactions discovered using CIE framework in turn help in measuring the connectivism learning in the classroom.

MELT - Models of Engaged Learning and Teaching

The Models of Engaged Learning and Teaching focus on students' thinking skills (Willison, 2018). Engaged teachers are vital for this process, and in this case, ChatGPT plays the role of a teacher. The six facets of MELT broadly address the skills used when researching, problem solving, critical thinking, evidence-based decision making, etc. "Optimizing Problem Solving" (OPS) is a model of problem-based learning that has been developed based on the MELT framework. The OPS model emphasizes the importance of creating authentic, complex, and ill-structured problems that require learners to engage in higher-order thinking, problem-solving, and decision-making (Missingham et al., 2018). The six facets of MELT-OPS are described below.

Embark & clarify (Find the purpose and clarify the learning outcomes): In the OPS model, this facet aligns with the "problem" element. As learners work on the problem, they clarify their learning outcomes and deepen their understanding of the subject matter (e.g., being able to correctly phrase questions).

Find & generate (Gather information and generate ideas): In the OPS model, this facet aligns with the "knowledge" element. Once learners have embarked on the problem-solving journey, learners gather, information, data and knowledge using the appropriate methodology (e.g., gather the code and explanations generated by ChatGPT).

Organize and manage (Develop a plan and manage resources): In the OPS model, this facet aligns with the "arrange" elements. Learners organise information & data to reveal patterns or themes. They also determine which prerequisite functions (basic, intermediate, advanced) are needed for completing more advanced tasks (e.g., gather additional information from ChatGPT and organize the information in a logical manner).

Evaluate & reflect (Review progress and revise understanding): In the OPS model, this facet aligns with the "trust" element. Once the learners have the information, they determine the credibility of sources, the information content and any data, and make their own decisions on the problem-solving methods (e.g., challenge the information given by ChatGPT).

Analyse & synthesise (Break down complex information and combine it into a new understanding): In the OPS model, this facet aligns with the "critical thinking" element. Learners analyse information/data critically and synthesise new knowledge to produce coherent individual/team understandings (e.g., validate the information from ChatGPT from classroom learning and external sources).

Communicate & Apply (Present and apply solutions): In the OPS model, this facet aligns with the "relate" element. As learners work through the problem, they will need to communicate their ideas effectively, apply their solutions to the problem and evaluate the effectiveness of their solutions (e.g., improve the answers given by ChatGPT and evaluate the solutions).

For the advanced programming course, our focus is on higher order cognitive taxonomy and the development of problem-solving skills. We study the students' interactions with ChatGPT based on the above MELT-OPS facets to understand the problem-solving skills of students in advanced programming.

Method

Computing Course Background

Text Mining and Language Processing is an advanced computing course under the undergraduate computing curriculum in our school. In this course, we cover topics in text mining including, document representation, text similarity, text categorization, text clustering, sentiment analysis, probabilistic topic models, and text visualization. Text mining techniques adopt the models from research areas such as Statistics, NLP, and Linguistics. We also focus on basic natural language processing techniques such as language parsing and analysis, information extraction, and evaluation techniques.

Study Group

For this study, we have 64 student participants who are enrolled in the course and completed both the lab review exercise as well as the survey. All the students are either in their year 3 or year 4 of study. 42% of participants are males and the remaining 58% are females. Most of the students (94%) are from the School of Computing and Information Systems and 6% from business and economics who either take computing as double degree or second major.

Data Collection

Lab Review Design

For programming lab assessment, we have two phases. In the first phase, we provide a lab exercise document for each week after the theory session to ensure students learn to apply text mining theory in real problems through python coding. The students complete the lab exercise by the end of the week and gain skills on advance programming. In the second phase, we create an in-class lab review exercise based on the previous three weeks' labs and students complete the lab review exercise using ChatGPT assistance. Some example lab questions include:

1. Write python code to choose the semantically correct syntax tree for the sentence "Mike coloured a kite on a tree".
2. Open K_means.py code file and go to the function compute_center (vecs, cluster). Explain the python code.

3. Analyse the below python code and fix the errors;

```
def corpus2docs(corpus):
    # corpus is a object returned by load_corpus that represents a corpus.
    fids = corpus.fileids()
    docs1 = []
    for fid in fids:
        doc_raw = corpus.raw(fid)
        doc = nltk.word_tokenize(doc_raw)
    docs2 = [[w.lower() for w in doc] for doc in docs1]
    docs3 = [[w for w in doc if re.search('^[a-z]+$', w)] for doc in docs2]
    docs4 = [[w for w in doc if w not in stop_list] for doc in docs3]
    docs5 = [[stemmer.stem(w) for w in doc] for doc in docs4]
    return docs5
```

Survey Design

To understand how students interact with ChatGPT, we designed survey questions that help to measure connectivism learning and problem solving skills. Table 1 shows the survey design for data collection and analysis. MELT levels indicate the student problem solving skills where he/she can make decisions on the answers provided by ChatGPT. Level 1 indicates that students can demonstrate basic problem solving skills and Level 2 indicates advanced problem solving skills. For example, under Find & Generate, "copying the code or getting the answer directly from ChatGPT" is considered a Level 1 problem solving skill whereas "asking ChatGPT to further explain the generated code" is considered a Level 2 problem solving skill. Level 2 skills indicate that students will produce high quality outputs and, in this case, more accurate lab answers. We first created the MELT-OPS survey questionnaire as it is a relevant model for Computing courses (SE MELT, 2023) and then aligned the questions with CIE. The questions are asked on a 5 point Likert scale (never, rarely, occasionally, regularly, always).

Question	**CIE Levels**	**MELT-OPS Facets**	**MELT Levels**
I rephrase the question to ChatGPT for more accurate answer	Operations	Embark & clarify	2
I just copy the code or the answer from ChatGPT	Wayfinding	Find & generate	1
I ask ChatGPT to further explain the code	Wayfinding	Find & generate	2
I ask ChatGPT to give other approaches or other codes	Wayfinding	Organise and manage	2
I challenge ChatGPT on the code or answer	Sensemaking	Evaluate & reflect	1
I challenge ChatGPT on the given explanations	Sensemaking	Evaluate & reflect	2
I validate the explanations with my own knowledge	Sensemaking	Analyse & synthesise	1
I validate the explanations with external knowledge (Internet)	Sensemaking	Analyse & synthesise	2
I choose the best from the different codes or answers given by ChatGPT by testing them.	Innovation	Communicate & Apply	1
I further improve the code or the answer generated by ChatGPT and submit lab review exercise.	Innovation	Communicate & Apply	2

Table 1. Survey design (MELT levels represent problems solving skill levels)

Level 1 for two facets, "Embark & clarify" and "Organise and manage", is ignored as level 1 for both facets is a basic problem solving skill that does not involve ChatGPT support. We also asked an open-ended question in the survey, "What are your takeaways from this lab session?" This question helps us understand the usefulness and any other sentiments that students experienced while using ChatGPT.

Data Analysis

For Likert scale survey questions, we first analyse the overall statistics on the questions. We then aggerated the results according to the CIE stages and MELT-OPS facets for a detailed analysis. Finally, we labelled the open-ended data into 4 categories; Positive, Negative, Caution and Others as shown in Table 2. This labelling helps to understand the sentiments of the students, the attitude of the students as well as the usefulness or challenges of the ChatGPT tool.

Comment	Label
ChatGPT is super useful for understanding concepts and getting a starting direction of how to solve a problem but is *currently not consistent* and <u>trustworthy enough to be blindly depended upon.</u>	**Positive**, *Negative*, <u>Caution</u>
Understanding concepts clearly is important	Others
The ChatGPT might generate different answer for the same question and we must decide if the answer is accurate	**Negative**, Caution
ChatGPT is a useful tool to get quick information, however it is important to verify it and I should not blindly copy the answer given by ChatGPT	**Positive**, Caution

Table 2. Human labeling of the student responses to an open-ended question

Findings and Analysis

In this section, we present statistics of ChatGPT which include student interactions, measurements of connectivism learning, measurements of problem learning skills and students' experiences with ChatGPT.

Interactions with ChatGPT

This section answers the research question; *how effective and engaging is ChatGPT in facilitating students' learning process*?

Overall Interaction Statistics

Figure 1 depicts the overall statistics of interactions with ChatGPT. Higher values for "occasionally", "regularly" and "always" (average of 71%) indicate strong interactions whereas higher values in "never" and "randomly" (29%) indicate weak interactions.

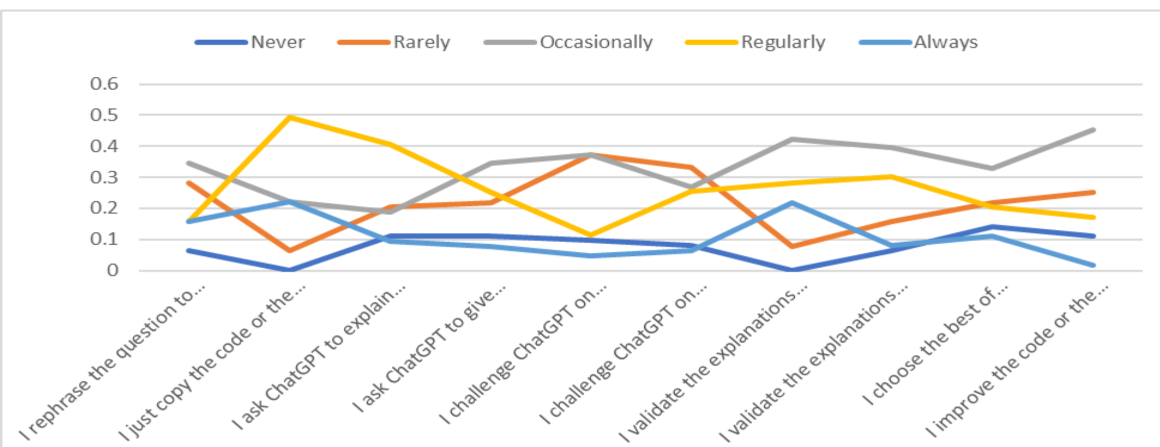

Figure 1: Overall statistics of interactions with ChatGPT

We also notice from the first two questions that some students do not put effort into rephrasing the question (34%) and most students do not check codes given by ChatGPT (71%). This can be due to the correctness of the code or students' trust in the tool. To understand this behaviour, we need to analyse student's behaviour towards the tool in terms of questioning its credibility. Based on the student answers, we notice on average only 17% of students question the credibility of the tool regularly or always whereas 35% of them check the credibility occasionally. To further analyse the engagement, we perform the interaction analysis using CIE and MELT-OPS frameworks.

Interactions Analysis by CIE Framework

Figure 2 shows the statistics of interaction by CIE framework and enables us to measure the connectivism learning of the students.

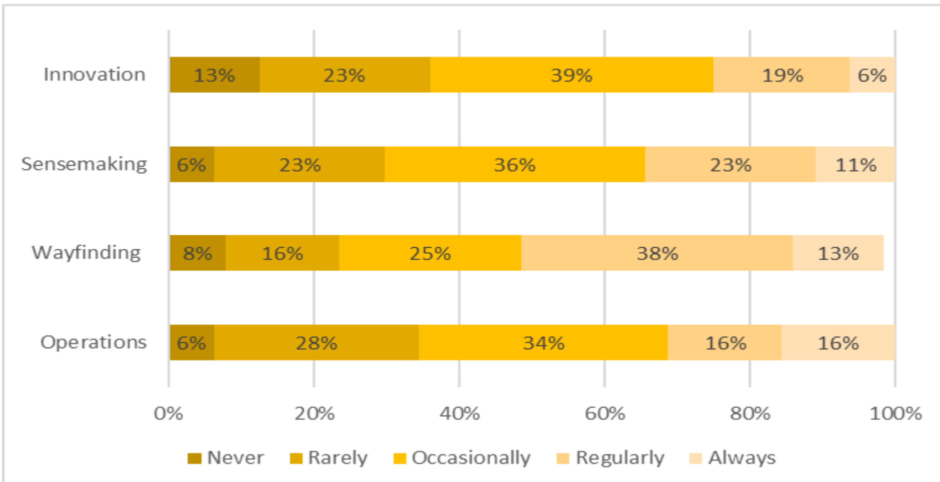

Figure 2: Statistics of interaction by CIE framework

During the lab:

Operation interaction occurred rarely to always (94%). This shows that the learners build and enhance their personalised learning environment with ChatGPT by asking and rephrasing questions.

Wayfinding interaction occurred rarely to always (92%). This shows that the learners interacted with ChatGPT for knowledge pipeline creation by asking questions to find the right information creating a simple connectivism learning.

Sensemaking interaction occurred rarely to always (94%). This shows that the learners interacted with ChatGPT for decision making based on the answers for the given lab problem. This stage has set a strong foundation for the next level namely innovation interaction by taking advantage of the collective knowledge from ChatGPT.

Innovation interaction occurred rarely to always (87%) but fewer times compared to the other interactions. This shows that the learners interacted with ChatGPT and external sources for obtaining their final lab solutions and testing them. At the same time, some students (13%) didn't go beyond the ChatGPT and were satisfied with the answers generated by ChatGPT.

Higher interactions between ChatGPT and learners at all levels of CIE framework indicate that connectivism learning is enabled by ChatGPT in the classroom. It is also effective in helping students gain advanced programming skills by providing them with an opportunity to carefully reflect and evaluate the answers given by the ChatGPT tool.

Interactions Analysis by MELT-OPS

Figure 3 shows the statistics of interaction by MELT-OPS framework and enables us to measure the problem-solving skills of the students. From the figure 3(a), we notice that some students (9% on average) do not use ChatGPT for problem solving skills in the labs. Whereas, an average of 70% of students exhibit problem solving skills occasionally. From Figure 3(b) we also notice that most of the students exhibit Level 1 skills ("never" for Level 1 is only 4% while "never" for Level 2 is 10%) to solve lab questions using ChatGPT. Similarly, "always" is 5% higher for Level 1 than Level 2 skills indicating that students exhibiting stronger Level 1 problem solving skills for various MELT-OPS facets.

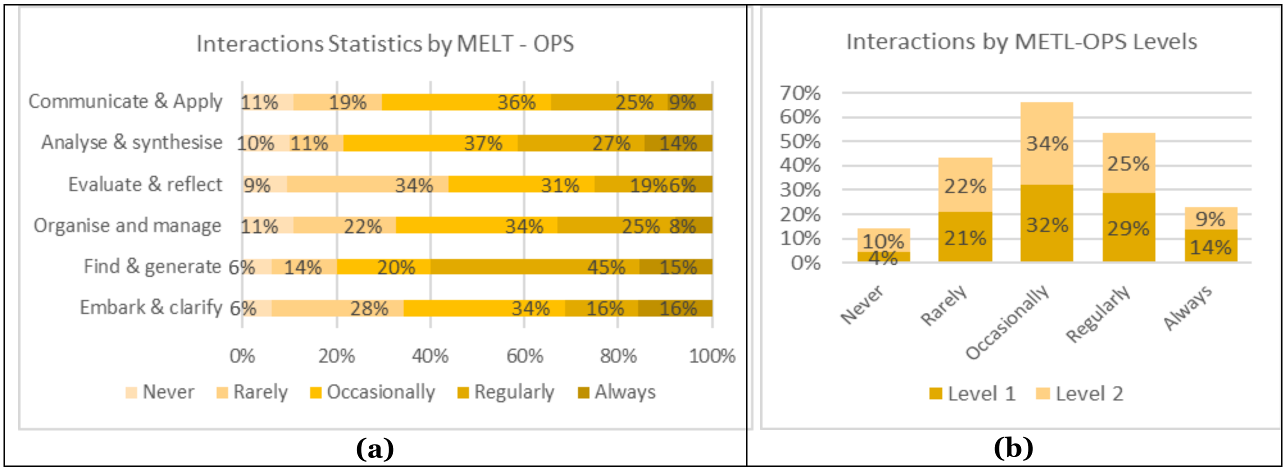

Figure 3: Statistics of interaction by MELT_OPS framework

The findings also indicate that the interactions between ChatGPT and learners are effective in learning advanced programming skills. However, from the CIE and MELP-OPS analysis we cannot know if the interactions were useful to the students. Hence, we use a textual feedback question in the survey to help understand the usefulness of ChatGPT in the lab.

Feedback analysis of ChatGPT

This section answers our second research question, *what are the experiences of students using ChatGPT as a support tool for learning advanced programming skills*. Figure 4 shows the sentiment statistics from the open-ended feedback question that helps to measure the usefulness of ChatGPT for the students for solving the lab problems. Positive sentiment indicates that the AI tool is useful. Negative sentiment indicated the AI tool was not useful. Additionally, we also added another dimension to the sentiments, caution, to study the students' attitudes toward ChatGPT.

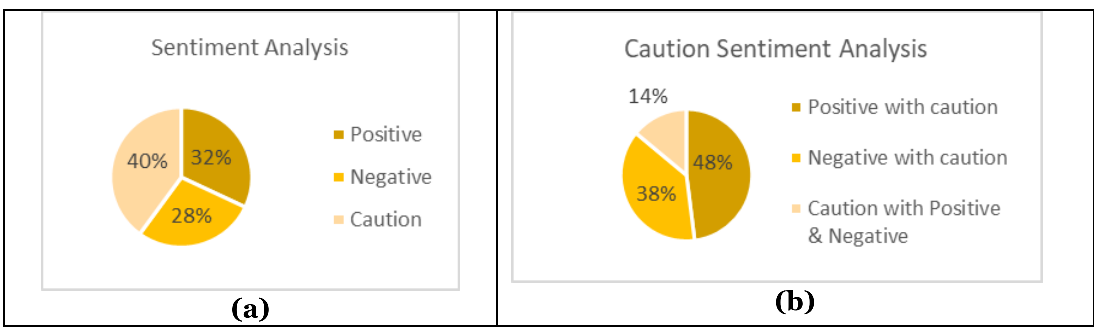

Figure 4: Sentiments statistics from the open-ended feedback on ChatGPT

The sentiment analysis from students' feedback indicates almost similar positive and negative sentiments about usefulness of ChatGPT. 32% of students find ChatGPT useful which is slightly higher that the negative sentiment of 28%. Table 2 shows some sample comments from students.

Positive Sentiments Samples	**Negative sentiments samples**
ChaptGPT is a good starting point, the code and the explanation is intuitive	chatGPT is not always correct or the most accurate
We should leverage the usage of ChatGPT when working	ChatGPT is not always right

in the future	
ChatGPT can highlight the different ways of doing the same thing	ChatGPT is not entirely accurate as compared to what is taught in class
ChatGPT is a very powerful AI chatbot	I dont like to use chatgpt :(
It has improved my understanding of the theory by enabling me to look at the code and how it works and its output	We might be better than ChatGPT and manual review is needed

Table 2: Example sentiments from students

Interestingly, 40% of the student indicated caution while using the tool. To further analyse the context of caution, we applied caution-based sentiment analysis and notice that around 62% of the caution comes with positive sentiments and positive as well negative sentiments. The students see the use of ChatGPT in learning and solving programming problems but at the same time are cautious about the answers. We also observe from negative with caution sentiments that 38% of students are unsatisfied with ChatGPT.

Limitations of Study and Future Work

Our study is a preliminary analysis of the ChatGPT in programming courses and has several limitations. The design of the experiments is a basic model, and we didn't study the A/B testing for this project. We also evaluated our learning model on 64 students which is relatively a small sample. Further, students may be from varying programming skill levels, and we did not analyse the pre and post programming skills levels and interaction levels to accurately measure the connectivism, problem solving skills and sentiments. For example, an expert programmer might be able to synthesise and evaluate the response from ChatGPT with more interactions while a non-expert programmer may not show such stronger problem-solving skills or connectivism learning. We also observed this issue from the answers to the open-ended question, such as "*Practise makes perfect*" and "*I will revise again*". Further, the prior knowledge of topics and time constraints might also have an impact on our studies. From one of the comments, "*Require better understanding of the materials taught in class, in order to process content taught in class and process the answers by chatGPT together quicker*", we observe the students' prior level of conceptual knowledge also impacts the interactions with ChatGPT. To understand the correlations between grades and interactions with ChatGPT, we studied the sentiments of the students against lab grades. We observed weak positive correlations between the student grades who mentioned cautions and positive sentiments towards the ChatGPT tool. Despite these limitations, we believe our results provide an initial indication that ChatGPT can be a successful learning tool for advanced programming courses.

Conclusion

In conclusion, the study shows that incorporating AI-powered chatbots, such as ChatGPT, in advanced programming courses can enhance student engagement and learning outcomes. Our findings indicate that the interactions between ChatGPT and learners enable connectivism learning in the classroom and can provide an effective learning tool for advanced programming. Although some students expressed a cautious attitude towards ChatGPT, the majority had positive sentiments, suggesting that chatbots can be a valuable support tool for learning advanced programming skills. Overall, this research provides initial evidence that ChatGPT can be an effective tool for supporting student learning in advanced programming courses. We believe that incorporating ChatGPT as a support tool in advanced programming courses not only benefits students but also helps instructors save time and focus more on students who require additional support in the class. Future studies could further investigate the potential benefits and challenges of using chatbots in programming courses to provide more comprehensive insights.

REFERENCES

Anderson, L. W., & Krathwohl, D. R. (2021). A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives. Longman.

Antaki, F., Touma, S., Milad, D., El-Khoury, J., & Duval, R. (2023). Evaluating the Performance of ChatGPT in Ophthalmology: An Analysis of its Successes and Shortcomings. medRxiv, 2023-01.

Belshaw, D. (2013). First draft of Mozilla's web literacy standard now available. Retrieved on July 26, 2015 from http://dougbelshaw. com/blog/2013/04/26/first-draft-ofmozillas-web-literacy-standard-now-available.

Chassignol, M., Khoroshavin, A., Klimova, A., & Bilyatdinova, A. (2018). Artificial Intelligence trends in education: a narrative overview. Procedia Computer Science, 136, 16-24.

Choi, J. H., Hickman, K. E., Monahan, A., & Schwarcz, D. (2023). ChatGPT Goes to Law School. Available at SSRN.

Chung, A. M., Bond Gill, I., & O'Byrne, I. (2017). Web Literacy 2.0. Repéré à: https://mozilla. github. io/content/web-lit-whitepaper.

Clear, A., Parrish, A. S., Impagliazzo, J., & Zhang, M. (2019, February). Computing Curricula 2020: introduction and community engagement. In Proceedings of the 50th ACM Technical Symposium on Computer Science Education (pp. 653-654).

Dishman, A. (2022). Teacher Perception of the Use of Twitter as a Professional Learning Tool (Doctoral dissertation, Southwest Baptist University).

Duke, B., Harper, G., & Johnston, M. (2013). Connectivism as a digital age learning theory. The International HETL Review, 2013(Special Issue), 4-13.

Dunlap, J. C., Sobel, D., & Sands, D. I. (2007). Designing for deep and meaningful student-to-content interactions. Tech Trends, 51(4), 20-31.

Englehart, J. M. (2009). Teacher–student interaction. International handbook of research on teachers and teaching, 711-722.

Laurillard, D. (1999). A conversational framework for individual learning applied to the 'learning organisation'and the 'learning society'. Systems Research and Behavioral Science: The Official Journal of the International Federation for Systems Research, 16(2), 113-122.

Missingham, D., Shah, S., Sabir, F., & Willison, J. (2018). Student engineers optimising problem solving and research skills. Journal of University Teaching & Learning Practice, 15(4), 8.

Okonkwo, C. W., & Ade-Ibijola, A. (2020). Python-Bot: A chatbot for teaching python programming. Engineering Letters, 29(1).

Riley, D. (2012). Using mobile phone programming to teach Java and advanced programming to computer scientists. In Proceedings of the 43rd ACM technical symposium on Computer Science Education (pp. 541-546).

SE MELT (2023). Software Engineering MELT Framework for Research. https://www.adelaide.edu.au/melt/ua/media/561/cs_sw_engineering_final_paper_2.pdf Retrieved on 20/4/2023.

Siemens, G. (2005). Connectivism: Learning as network-creation. ASTD Learning News, 10(1), 1-28.

Siemens, G. (2012). Orientation: Sensemaking and wayfinding in complex distributed online information environments (Doctoral dissertation, University of Aberdeen).

Utecht, J., & Keller, D. (2019). Becoming Relevant Again: Applying Connectivism Learning Theory to Today's Classrooms. Critical Questions in Education, 10(2), 107-119.

Verleger, M., & Pembridge, J. (2018, October). A pilot study integrating an AI-driven chatbot in an introductory programming course. In 2018 IEEE frontiers in education conference (FIE) (pp. 1-4). IEEE.

Vihavainen, A., Paksula, M., & Luukkainen, M. (2011, March). Extreme apprenticeship method in teaching programming for beginners. In Proceedings of the 42nd ACM technical symposium on Computer science education (pp. 93-98).

Wang, Z., Chen, L., & Anderson, T. (2014). A framework for interaction and cognitive engagement in connectivist learning contexts. International Review of Research in Open and Distributed Learning, 15(2), 121-141.

Whitney, M., Lipford-Richter, H., Chu, B., & Zhu, J. (2015, February). Embedding secure coding instruction into the ide: A field study in an advanced cs course. In Proceedings of the 46th ACM Technical Symposium on Computer Science Education (pp. 60-65).

Willison, J. W. (2018). Research skill development spanning higher education: Critiques, curricula and connections. Journal of University Teaching and Learning Practice, 15(4), 1.

Zhai, X. (2022). ChatGPT user experience: Implications for education. Available at SSRN 4312418.