CS846 Week 2: How Far Are We? The Triumphs And Trials of Generative AI in Learning Software Engineering

Mohammad Jaffer Iqbal

Problem Being Solved:

This paper highlights a knowledge gap when it comes to using conversational generative AI for learning advanced CS topics like Software Engineering. Existing literature focuses on Introductory CS courses in a learning context using conversational GenAI. This problem can then be split up into two research questions: (i) How effective is conversational GenAI in helping students in software engineering tasks and (ii) What are the current shortcomings of GenAI for learning complex CS topics like SE.

New Idea:

To answer the first research question, the paper chooses to perform an experiment with a treatment group of students only using ChatGPT (a representative conversational GenAI model) to aid them in solving SE tasks (consisting of debugging, removing code smells, and collaborating using Github) while the Control group had access to traditional non GenAI resources. The students then filled questionnaires based on AAR/AI, Microsoft's HAI guidelines, and how they felt about their experience. Their performance in the tasks along with their responses to the questions were then used to perform a statistical and qualitative analysis to highlight performance differences and the pitfall of ChatGPT in the experiment.

Positive Points:

The paper includes a very well structured and detailed methodology consisting of well thought out Hypothesis and choice of metrics to prove/disprove the hypothesis. Next, the paper did well to highlight the perceived violations of HAI guidelines by ChatGPT and discuss their underlying reasons and the question and implications. Finally, the paper also included a very informative future works section to address the highlighted pitfalls.

Negative Points:

The paper assumes Python tasks as being representative of SE tasks, which is not the case. Software Engineering includes working with large codebases (the tasks pertained to a small codebase), containing complex and intricate bugs that may have unwanted effects (the tasks had small, isolated bugs) along with containing new and different frameworks. Furthermore, the paper only focuses on development tasks and not those pertaining to Software Engineering which includes planning and requirement design. The sample size of just 22 students raises questions about the generalizability of the results. These 22 students were chosen from different Software Engineering courses. While this choice was made to ensure larger coverage and diversity, it also means that the students had different levels of expertise which may interfere with attributing the students' performance to the tools they used. Finally, the paper bases its results of a single session; however, the results of the study may change if it spanned multiple sessions.

Future Work:

Apart from the future works mentioned in the paper, one concrete direction that could be explored would be to have a conversational GenAI that trains using conversation history and understands the user's learning style. Then, performing an experiment with this model in a Treatment group and having another GenAI model without such a feature in the Control group would help understand if figuring out the learning style of a user helps provide better assistance and avoid some of the identified pitfalls. Furthermore, the experiment could be conducted in a co-op setting which would have tasks with the complexity of those in the industry. The study could also be performed over a longer time (like a term) to understand differences in productivity.

Rating:

4.5 – The paper contained a very well detailed experiment containing clear reasonings for design choices made. The authors have also open sourced their prompts and contains extensive threats to validity.

Discussion Points:

(i) Which pitfalls (limited advice, inability to comprehend problems, incomplete assistance, hallucination, wrong guidance) have students personally experienced while using ChatGPT for SE related tasks? **Discussion:** Students have experienced several pitfalls using ChatGPT for SE related tasks. While it excels at explaining concepts, it struggles with heavy context and vague prompts. Debugging is another challenge: without sufficient context, ChatGPT provides only surface-level suggestions, which is problematic since users often lack full bug details. Additionally, it is good at explaining documentation but lacks context retention, making it less effective for ongoing tasks.

(ii) Are there any potential Issues with the Experiment design? **Discussion:** There is a gap between the expertise of first-year CS students and fourth-year students. First-year students may not fully benefit from ChatGPT if they lack the background knowledge to ask "useful" questions or interpret feedback from ChatGPT. Even though the experiment does not recruit first-year students, they do include second-year students to which the aforementioned concerns apply. Advanced learners know the specific area or part of the code they want help with, so they can get more value from ChatGPT's explanations or code suggestions. Furthermore, the group that used ChatGPT actually had less choice as the only access they had was ChatGPT, which needed some familiarity with the problem to solve to use. The treatment group had more diverse choices as they could access Google and StackOverflow etc. It would be better if they could provide some extra information like a guide for using ChatGPT to the experiment group.

(iii) Does using GenAI in SE adversely impact a student's independent debugging capabilities? **Discussion:** This depends on the way a student uses ChatGPT to debug. If they provide it with the code and a general prompt for debugging, it will increase their over-reliance on ChatGPT and reduce their capabilities over time. However, if the student is keen on understanding the problem without blindly relying on ChatGPT for debugging, then it can become an aid instead of being a detriment over time. As long as students identify the specific portion of code or logic that is troublesome to get a good response from ChatGPT, it may not negatively impact their capabilities that much. It is always a good approach to verify the answers from GitHub, Stack Overflow, or official documentation.

(iv) If an experiment is conducted with a group of students that can use a mixture of ChatGPT + traditional resources -> how do you expect this group to perform? **Discussion:** Such a scenario would not be helpful to prove or disprove the current hypothesis. However, another study in which a treatment group contains clear guidance on how to use ChatGPT in conjunction with other sources while the Control group uses only ChatGPT as their aid would clarify the pitfalls of the current study design (pertaining to the students feeling constrained with just using ChatGPT).