CS 846 Software Engineering for Big Data and AI

On the assessment of generative AI in modeling tasks: an experience report with ChatGPT and UML

Jun Lim

20870249

The rise of generative AI and large language models (LLMs) such as GitHub's Copilot and OpenAI's ChatGPT has created a disruption in the software industry. While LLMs excel in code writing, their proficiency in software modeling remains unexplored. The authors aim to explore the potential of these models in conceptual software modeling while also trying to identify their shortcomings.

The authors conducted the experiment in the following phases:

1. **Exploration phase**: authors interacted individually with ChatGPT to understand the modeling capabilities. This is done by using the prompts of ChatGPT to create models of different sizes to determine the features and limitations of ChatGPT in modeling tasks (such as creating UMLs).
2. **Focused phase**: the authors conducted a more systematic and focused approach to evaluate the ChatGPT's prompt outcomes of various modeling concepts and mechanisms (such as classes, attributes, generalization, etc.).

The outcomes are then obtained and analyzed with the following findings:

1. ChatGPT generally generates accurate UML models but contains some small syntactic errors.
2. ChatGPT often suggests models that are not related to the modeling aspect.
3. ChatGPT's results heavily depend on the problem domain, where accuracy depends on the amount of information it knows. It also performs badly when the names of entities lack meaning or reference.
4. ChatGPT is unable to handle models with a large number of classes.
5. ChatGPT handles some modeling concepts well while others are not; it requires explicit ”indication of concept.”
6. ChatGPT's prompt results vary for each conversation (inconsistent response).
7. ChatGPT's modeling is iterative; it requires starting with a basic model and adding details; its inconsistent responses often require restarting conversations for better accuracy.
8. ChatGPT produces better models with OCL notation than UML.

The paper then finally concludes that ChatGPT is not yet a reliable tool to perform modeling tasks. Its performance is limited in comparison to code generation and completion. The authors also discussed the potential role of LLMs in software modeling and suggested ways to enhance the capabilities and future of LLMs for software modeling.

Paper Commentary

The paper provided fresh perspectives on the usage of ChatGPT in a new domain of software modeling. It also made a valuable contribution to the software engineering community by bridging the gap between AI and software. The depth of research conducted was decent, as it involved a methodological and systematic approach to providing insights. Overall, the paper was well-written and well-structured, with distinct sections dedicated to introduction, context, experiments, findings, and conclusion, making it accessible to readers from various backgrounds.

Given the inherent variability of ChatGPT's responses, reproducing the exact response might be challenging for researchers. Thus, it is unsure how consistent the findings would be on a broader scale of things. The authors also could have added broader discussions on the practical benefits of using LLMs in software modeling, as the idea of software modeling with ChatGPT seems to have little value in the real world.