

For Modeling Programmers as Readers with Cognitive Literary Science

Rijul Jain
Williams College
rijul.jain@williams.edu

Abstract

The prevalent text-based form of human-generative AI interactions has blurred the lines between code and prose. I argue that understanding the cognitive effects of these interactions by modeling users not only as programmers, but as readers, will inform the design of better tools to bolster human agency in generative AI interactions. I hope to begin a conversation around the uses of cognitive literary science for the study of the psychology of programming.

Reflections

The analogy between the “complexity of a large and thoughtful literary novel” and that of “a large computer program” holds because both literature and computer programs deal in “sophisticated information representations” (Blackwell, 2023). The forms of these representations change—especially now, as using natural language to interact with generative AI in increasingly formalized ways (or, prompt programming) has created a new way to conceive the forms that programming can take. Whether they have ever written code or not, users of generative AI interfaces are therefore often programming in some respect—but they are simultaneously put in the position of readers, reacting to and being acted upon by the varied, sometimes surprising outputs given by, for example, ChatGPT, while also directing the series of interactions to write the narrative of the exchange.

Questions of improving human agency with respect to new programming notations and generative AI have therefore never been more keenly related to developing and learning from cognitive models not only of programmers, but of readers working upon and being worked upon by texts. At PPIG 2023, Lewis (2023) indirectly took up these concerns by inquiring whether the “psychology of natural semantics” would “become a central part of the psychology of programming,” with a renewed focus on studying end-user programming with LLMs. Yet these issues bear on expert programmers, too—Floyd et al. (2017) find in an fMRI study of programmers’ brains that experts “treat code and prose more similarly at a neural activation level” than non-experts.

Hermans et al. (2017) frame programming and writing as closely related. To understand generative AI users as programmers even more comprehensively, it will also prove fruitful to model them as readers by attending to cognitive literary science—research using cognitive psychology and neuroscience to explicate the mental processes and the corner cases of reading. Gerrig et al. (2003) lay out readers’ processes of continually updating mental representations of narrative experiences—their work shows literature to be a fertile ground for throwing into relief cognitive processes relevant to the psychology of programming. In particular, Bergs’ (2017) research on coercion, or “the resolution of formal mismatch,” uses examples from literature to illustrate and complement neuroscience work on a fundamental cognitive phenomenon—one crucial to understanding how users react to unassimilably and unrectifiably anomalous output from generative AI.

More generally, modeling programmers as readers will advance research directions aimed at designing for diverse programming notations and bolstering human agency. Crichton et al. (2021), for example, draw from cognitive psychology—evaluating the load on programmers’ working memory during program tracing—to inform conclusions about programming tool design. Following this approach with cognitive literary science to understand the cognitive effects of programmers’ higher-level, often narrativized generative AI interactions may similarly reveal patterns and pitfalls to inform better design for generative AI tools. Doing so even more broadly will also improve computational creativity—e.g. Chandra et al., (2023) who create a new framework for animation in part by modeling their audience with cognitive-science-informed narrative theory. Efforts at computational creativity going forward must be attuned at a cognitive level to the ways people experience the arts.

Current forms of text-based interaction with generative AI systems have blurred the lines between code and prose. As their forms of representation tend toward convergence, and if indeed “AI is a branch of literature,” (Blackwell, 2023) understanding and evaluating how programmers-as-readers interact with speech in this generative AI context will propel the development of more reliable and usable technical notation than natural language alone. Insights from cognitive literary science may then yield a fuller picture of the possibilities of enabling human agency with respect to programming.

References

- Bergs, A. (2017). Under Pressure: Norms, Rules, and Coercion in Linguistic Analyses and Literary Readings. In Burke, M., and Troscianko, M. T. (eds), *Cognitive Literary Science: Dialogues between Literature and Cognition*, (Oxford: Oxford University Press).
- Blackwell, A.F. (2023). Chapter 14: Re-imagining AI to invent more Moral Codes. Retrieved from: <https://moralcodes.pubpub.org/pub/chapter-12/release/4>
- Chandra, K., Li, T., Tenenbaum, J. & Ragan-Kelley, J. (2023). Acting as Inverse Inverse Planning. In ACM SIGGRAPH 2023 Conference Proceedings (SIGGRAPH '23), 7, 1–12. <https://doi.org/10.1145/3588432.3591510>
- Crichton, W., Agrawala, M., & Hanrahan, P. (2021). The Role of Working Memory in Program Tracing. In Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21), 56, 1–13. <https://doi.org/10.1145/3411764.3445257>
- Floyd, B., Santander, T., & Weimer, W. (2017). Decoding the Representation of Code in the Brain: An fMRI Study of Code Review and Expertise. 2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE), (175-186). doi: 10.1109/ICSE.2017.24.
- Gerrig, R. J., & Egidi, G. (2003). Cognitive psychological foundations of narrative experiences. In D. Herman (Ed.), *Narrative theory and the cognitive sciences*, 33–55. Center for the Study of Language and Information.
- Hermans, F. & Aldewereld, M. (2017). Programming is Writing is Programming. In Companion Proceedings of the 1st International Conference on the Art, Science, and Engineering of Programming (Programming '17), 33, 1–8. <https://doi.org/10.1145/3079368.3079413>
- Lewis, C. (2023). Large Language Models and the Psychology of Programming. In Proceedings of the 34th Annual Conference of the Psychology of Programming Interest Group (PPIG 2023), 77-95.