

# The “Expert Tool” Trap: Notes on an AI-assisted Methods Paper

Brett Reynolds 

Humber Polytechnic & University of Toronto

[brett.reynolds@humber.ca](mailto:brett.reynolds@humber.ca)

10th January 2026

## 1 INTRODUCTION

This document reflects on the process of writing *Grammatical Variation Meta-Study* with an AI agent. The initial goal was efficiency: delegate the “boring” parts of Bayesian modeling and LaTeX formatting to a machine. The reality was less about acceleration and more about collision. The agent did not simply implement a pre-existing plan; it exposed the fragility of that plan through its own failures.

This log details three specific crises—technical, epistemic, and rhetorical—that forced a deviation from the standard research workflow. It suggests that the value of AI in research is not as a *force multiplier* but as an *adversarial partner*.

## 2 THE “EXPERT TOOL” TRAP (TECHNICAL)

The most dangerous moment in the project occurred on January 10, when I attempted to fit a joint selection–outcome model using the industry-standard package `brms`.

The logic was sound: I needed to model the probability of a variable being `TESTED` ( $y_1$ ) and, conditional on that, the probability of it being `FOUND` ( $y_2$ ). A multivariate model seemed the obvious choice. The agent wrote the code, the model compiled, the chains converged ( $R < 1.01$ ), and estimates were precise.

They were also wrong.

As noted in the pedagogical log (2026-01-10 *Multivariate brms Missingness Check*), the `brms` package, essentially an expert system for regression, assumes by default that missing values in outcomes should be dropped (listwise deletion). Because  $y_2$  is undefined for untested variables, the package silently dropped all rows where  $y_1 = 0$ . The resulting model was mathematically valid but scientifically vacuous: it modeled selection only among those already selected.

The agent, trained to produce “correct” code, produced code that was syntactically perfect but semantically fatal. The fix was not better prompting. It was a retreat to first principles: writing a custom Stan model to explicitly define the likelihood for the untested case. The agent could write the Stan code, but only after the human identified that the “easy” path was a dead end.

### 3 SIMULATION AS CRITIQUE (EPISTEMIC)

The second crisis was conceptual. The early drafts of the paper framed the high success rate of grammatical variables as evidence of “human sociolinguistic capacity.” It was a celebration of the field’s ability to find meaning everywhere.

To test this framing, I ran a *Simulated Reviewer* session (2026-01-10 *Simulated Reviewer Critiques*), generating feedback from personas representing distinct epistemic communities (e.g., *Zimmer* for variationist theory, *Gelman* for statistics).

The critique was blistering. The *Godfrey-Smith* persona pointed out a fundamental confusion between the map and the territory:

“You are confusing the capacity of the organism with the filter of the institution. A high success rate in published papers doesn’t mean language is saturated with social meaning; it means researchers are rational actors who don’t publish nulls.”

This was not a copy-edit. It was a refutation. It forced a rewrite of the Abstract and Discussion (2026-01-10 *Drafting Learnings Reflection*), shifting the claim from a linguistic discovery to a sociological one. The AI, usually a compliance machine, became a mechanism for estrangement, allowing me to see the paper’s flaws through an outsider’s eyes before they reached a human reviewer.

### 4 RHETORIC AS DEBUGGING (SYNTHESIS)

The final lesson was that code and prose are not separate domains. In trying to explain the selection model to a general audience, I found myself relying on lazy academic connectives: *Thus*, *Therefore*, *It is worth noting*.

The agent’s enforcement of “plain style” (2026-01-10 *Plain Style Enforcement*) revealed that these connectives were often masking logical gaps. If a sentence couldn’t stand without a *Therefore*, the logical connection usually wasn’t in the data.

I replaced these with classical rhetorical figures. For example, to sharpen the distinction between the two stages of the model, I used SYMPLOCE (repeating the start and end of clauses):

- *Selection asks*: Among all possible variables, which are chosen for study?
- *Outcome asks*: Among those chosen, which are reported as significant?

This was not decoration. It was debugging. The constraint of the rhetorical form forced a precision in the conceptual structure that the code had missed.

## 5 CONCLUSION: THE “SUDO” BOUNDARY

The project concluded with a minor operational check: the agent paused before running a sudo command (2026-01-10 *Privileged Actions Check-In*). This boundary remains the best metaphor for the collaboration. The agent can suggest, code, simulate, and critique. But it cannot start the machine, and it cannot sign the paper.

The efficiency narrative promises that AI will do the work. The reality of this project was that AI created *more* work—more code to check, more critiques to answer, more drafts to refine. But it was better work. The paper is more rigorous, more humble, and more true than the one I would have written alone.