Discussion of "Task Efficiency and Signaling in the Age of GenAI: Effort Reallocation and Firm Value Effects" by Shiwei Ye

Disscussed by Miao Ben Zhang

USC Marshall School of Business

2026 ASSA Meeting

Background: In need of GenAI's de facto effects

Generative AI is widely regarded as a ground-breaking new technology, triggering massive interest for research to understand its impact on workers and firms. Current approaches:

- Research using lab experiment or survey
  - e.g., Kosmyna et al. (2025), Bick et al. (2024), Humlum and Vestergaard (2025)...
  - Lab: Limited in speaking to workers' *overall responses* in task allocation
  - Survey: Limited in tracking workers' task allocation *over time*
- Research examining natural experiment within a firm
  - Brynjolfsson et al. (2025) and others
  - Clean effects but firm/occupation-specific and likely short-termed
- Research studying de facto impact on workers and firms
  - Eisfeldt et al. (2025), Jiang et al. (2025), Sheng et al. (2025), Kogan et al. (2025), Hampole et al. (2025)...

This paper: Combining an event-study setting with detailed data to examine heterogeneous de facto impact of GenAI on workers

Research Question & Context

Question: How does Generative AI reshape effort allocation when it simultaneously:

1. Increases productivity in AI-assisted tasks
2. Erodes signaling value of those same tasks

The Puzzle:

- Lab studies: GenAI helps juniors most (Brynjolfsson et al. 2023; Noy & Zhang 2023)
- Real world: Entry-level hiring *declined* in AI-exposed occupations (Brynjolfsson et al. 2025)
- Why? Employers struggle to assess workers' genuine ability without AI

This Paper: Studies effort reallocation between coding (AI-assisted) and innovation (human-centric) using open-source software development

This Paper

Empirical laboratory:

- **Sample:** 26,026 GitHub developers from 1,281 U.S. public firms (2021-2023)
- **168,085 repositories:** Both coding activity & innovation outcomes observable
- **Identification:** DiD using GitHub Copilot Launch (June 21, 2022)

AI exposure measurement for workers and firms:

- Language-level AI scores via GPT-3.5: Python=1.0, JavaScript=0.9, Stata=0.5
- Aggregated to developer-level and then firm level.

Main results:

- Worker-level: Heterogeneous effects by seniority
  - Seniors: coding activities rise; project value declines; no effect on labor market
  - Juniors: increased project value and novelty; more job moves and promotion for innovators.
- Firm-level: Matching of firm-innovativeness and worker-seniority matters, through stock market reaction to copilot launch

Bottomline: Career concerns matter in real world (vs. short-term experiments)

Overall View

There is a lot to like about this paper

1. New Mechanism for AI & Inequality: on seniority-biased effects
2. Firm Value & AI Adoption highlighting employer-employee alignment
3. Methodological Contribution: LLM-based language exposure

My comments focus on tightening the paper

Comment 1: 2SLS missing the first stage

The issue:

- Paper wants to say:

  AI exposure $\rightarrow$ Copilot adoption $\rightarrow$ outcomes

- The test in the paper:

  AI exposure $\rightarrow$ $\rightarrow$ outcomes

- Is channel of the effects through **Copilot adoption**?
  - A Python expert might NOT USE Copilot (personal preference, firm policy)
  - A Stata expert might USE Copilot extensively for their secondary Python work
  - Validation at the language level $\neq$ 1st-stage test at the developer level

- Challenge: GitHub developers do not disclose whether they used Copilot

Comment 1: 2SLS missing the first stage

Suggestions: here is what I got after brainstorming quite a bit

1. Direct Survey of Developers on GitHub
   - "Did you use GitHub Copilot for this project?"
   - Helpful, costly, and subject to survey bias

2. Code Similarity
   - Use tools like `code2vec` or GitHub's CodeQL to detect: (i) Similarity to common AI-generated patterns; (ii) Boilerplate code quality improvements, etc.

3. Telltale Signs of AI Assistance in Codes
   - Generic/templated commit messages
   - Sudden changes in commit message style
   - Comments like "Co-authored-by: GitHub Copilot"
   - More descriptive docstrings (AI generates better documentation)

Challenging to find perfect validation, but evidence on 1st-stage is welcome and helpful.

Comment 2: Timing of treatment

Timeline:

- June 29, 2021: GitHub Copilot was released as a technical preview
- June 21, 2022: Copilot was officially launched

  "With more than 1.2 million developers in our technical preview over the last 12 months, people who started using GitHub Copilot quickly told us it became an indispensable part of their daily workflows. " *GitHub Blog June 21, 2022*

The paper uses June 21, 2022, the official launch of GitHub Copilot, as the event time

- Is it possible that younger developers are more likely to have already incorporated Copilot before the official launch?
- The heterogeneous effects by seniority is driven by timing of adoption rather than heterogeneous task allocation

Suggestion:

- Do the results reverse or disappear after a longer period post event time?
- Again, it would be helpful to see who adopts Copilot

Comment 3: Many important possibilities to explore

1. **External Validity:**
   - How well do GitHub signaling dynamics generalize to internal firm activities?
   - Are open-source contributors representative of broader developer population?
2. **Long-Run Equilibrium:**
   - Will markets eventually develop new screening mechanisms?
   - How persistent is the signal dilution effect?
3. **Firm Strategy:**
   - Optimal workforce composition in equilibrium?
4. **Welfare:**
   - Net welfare effects accounting for productivity gains vs. signaling costs?
   - Social value of innovation vs. coding efficiency?

Conclusion

- An important study on how GenAI affects young and senior developers
- Novel findings on employer-employee alignment (at least in the short run)
- Many important implications to explore
- My comments focused on tightening the mechanism tests
- Highly recommend!