GitHub

88 Colin P Kelly Jr Street
San Francisco, CA 94107
United States of America

November 17, 2023

RE: Implications of Artificial Intelligence Technologies for the Canadian Labor Force

Thank you for the invitation to testify before the House of Commons' Standing Committee on Human Resources, Skills and Social Development and the Status of Persons with Disabilities (HUMA) in order to inform your study on the Implications of Artificial Intelligence Technologies for the Canadian Labor Force. Below I offer brief written remarks to support your study, in my role as Senior Policy Manager at GitHub.

GitHub is the home for all developers and the center of global software innovation. We operate a code hosting and collaboration platform for more than 100 million developers globally and offer AI-powered software development tools to individuals and organizations. Most notably among these is GitHub Copilot, our AI-powered pair programming application which has over 1 million paying users.

In Canada, 2.26 million developers contribute to public code on GitHub, a 99% increase from two years earlier. They collaborate on software projects around the world, including 6.28 million projects based in Canada.[1] These developers contribute to innovation in Canada, and increasingly they are empowered with AI tools.

As part of our commitment to developing and deploying AI tools responsibly, GitHub has conducted productivity research and supported academic collaborations to understand the possible implications of AI-powered code generation tools on the software development labor force and open source software communities.

Developers can write software up to 55% faster using GitHub Copilot. This result is based on a randomized controlled trial detailed in an openly available paper.[2] Although real-world settings may see different productivity jumps–a subsequent observational study of some 1 million Copilot users found a figure closer to 30%[3]–the task of writing software code has been greatly accelerated by AI.

---

[1] Precisely, these projects have more collaborators from Canada than any other country. https://innovationgraph.github.com/economies/ca
[2] https://arxiv.org/abs/2302.06590
[3] https://arxiv.org/abs/2306.15033

Considering jobs as bundles of tasks, the increased speed at writing software leaves developers more time for other tasks, or to get more ambitious by learning new programming languages that support cybersecurity via memory safety[4] or that are better suited for particular tasks.[5] Developers using GitHub Copilot report increased wellbeing as a result: 60% reported increased job satisfaction and 74% reported being able to focus on more satisfying work.[6] This shift can be viewed as a continuation of transformation in the software development industry since its advent, of automation enabling higher-order thinking by lessening the tedious, repetitive work developers often perform. The compiler, programming languages that abstract away details, open source software that allows others to build with a world of preexisting components, integrated development environments, and code scanning and testing all changed the ways that developers build software. Amid those transitions, it is important to note that, based on US statistics, costs fell for software, demand increased as digital transformation accelerated, and the number of developers and their pay both increased as a result.[7]

Our productivity research on GitHub Copilot found that developers who were less experienced and older in age benefitted the most.[8] Subsequent studies on other generative AI tools in professional writing[9] and service[10] tasks similarly found that relative novices benefited most. In software development, there is reason to hypothesize that such tools can enable more people, including those from disadvantaged and/or atypical backgrounds, to begin careers in software development. GitHub is pursuing programming[11] and academic research to support and measure precisely this.

AI-powered developer tools are some of the most widely adopted professional AI tools today: surveys indicate between 70-92% of developers are using AI tools at work or plan to begin doing so.[12] As such, they may offer a preview of

---

[4] https://github.blog/2023-08-30-why-rust-is-the-most-admired-language-among-developers/#why-developers-love-rust

[5] An anecdote from a professor who used GPT-4 to learn the Lean 4 programming language https://news.ycombinator.com/item?id=38035672

[6] https://github.blog/2022-09-07-research-quantifying-github-copilots-impact-on-developer-productivity-and-happiness/

[7] Footnote 1 in https://www.brookings.edu/articles/how-ai-powered-software-development-may-affect-labor-markets/

[8] https://arxiv.org/abs/2302.06590

[9] https://www.science.org/doi/10.1126/science.adh2586

[10] https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4573321; https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4426942

[11] https://allinopensource.org/

[12] A recent survey of professional software developers in the United States found that 92% are using AI code generation tools at and outside of work: https://github.blog/2023-06-13-survey-reveals-ais-impact-on-the-developer-experience/. A larger-scale, international survey found that 70% of developers

the future of work and wider policy implications as further occupations adopt AI tools. Drawing on a recent policy report,[13] I highlight three below:

1. **Support lifelong learning for professionals.** More occupations may come to resemble software development, and in particular its emphasis on learning new skills. The skills employers expect of developers changed more than 30% between 2015 and 2022.[14] Policymakers and employers alike should support workers in lifelong learning, in the short-term encouraging employees to experiment with AI tools and to share what they have learned, in brainstorms, hackathons or other informal venues. Over time, best practices on how to teach these tools will emerge, and policymakers should encourage employers to provide time and support employees in learning these AI skills.

2. **Invest in digital equity education.** AI may enable more people to write code, but digital skills remain valuable as ever. The mental model of articulating a software specification or prompting for specific functions will remain important. To prepare the next generation of software developers and others with essential digital skills, policymakers should continue to focus on basic digital equity like broadband internet access, computer science in primary education, and additional opportunities to use AI tools in a supervised manner.

3. **Monitor labor market impacts over time and prepare for contingencies.** Although the productivity effects and labor market impacts may take time to become apparent at a large scale, the capabilities of AI tools are expected to continue to advance. Small-scale, controlled experiments on the productivity of new tools can complement more speculative, forward looking projections of job impacts[15] to provide policymakers with leading indicators of job impacts to inform policymaking for targeted services. As AI tools become integrated into more professional workflows, the possibilities for rapid change may increase as more highly capable AI models can be easily swapped into existing workflows. The possibility, however remote, of discontinuous jumps in task-specific and possibly labor automation leads some experts to emphasize the need to plan for contingencies requiring structural reforms to tax and benefits schemes.[16]

---

are using or plan to begin using AI tools in their software development: https://survey.stackoverflow.co/2023/#ai-developer-tools.

[13] https://www.brookings.edu/articles/how-ai-powered-software-development-may-affect-labor-markets/

[14] https://linkedin.github.io/future-of-skills/

[15] https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4414065; https://arxiv.org/abs/2303.10130

[16] https://direct.mit.edu/daed/article/151/2/272/110622/The-Turing-Trap-The-Promise-amp-Peril-of-Human; https://www.nber.org/papers/w30172

As with other general purpose technologies before it, AI will bring significant benefits as it is adopted across the economy. We welcome your committee's focus on AI's implications for the Canadian labor force, to address possible risks so that we can collectively seize the benefits. Software developers are early adopters of these technologies, and, thus, can offer a leading indicator for the future of work. I hope that the research and papers presented above can help you in your important study, and please do not hesitate to reach out with any questions.

Sincerely,

Peter Cihon

Senior Policy Manager
GitHub