THE
QUARTERLY JOURNAL
OF ECONOMICS

Vol. 141 2026 Issue 3

THE POWER OF PROXIMITY TO COWORKERS*

Natalia Emanuel
Emma Harrington
Amanda Pallais

How does proximity to coworkers affect training and productivity? We study software engineers at a Fortune 500 firm from 2019 to 2024, leveraging two shocks to proximity: the office closures in 2020 and the subsequent return-to-office mandates in 2022 and 2023. In both cases, co-located teams experienced bigger changes in proximity than distributed ones, facilitating difference-in-differences designs. We find that sitting near teammates increases coding feedback by 18.3% and improves code quality. Gains are concentrated among less-tenured and younger employees, who are building human capital. However, there is a trade-off: experienced engineers write less code when sitting near teammates. In national U.S. data, we find evidence that the rise of remote work has had scarring effects on young college graduates. In remotable jobs, young graduates' unemployment rate increased

* This project would not have been possible without the curiosity and commitment to research of our colleagues at the firm who shared data: Lauren, Susan, and Ebube. We thank Alex Albright, Jenna Anders, Eric Anderson, Garrett Anstreicher, Christopher Campos, Raji Chakrabarti, Kirsten Clinton, Ed Glaeser, Ben Hyman, Bill Johnson, Ezra Karger, Lawrence Katz, Gizem Kosar, Sitian Liu, Daniel Mangrum, Ishan Nath, Dev Patel, Soum Shukla, Jason Somerville, Mattie Toma, Tianyi Wang, and Melanie Wasserman for thoughtful comments. We thank numerous seminar participants. We thank Jared Fang for research assistance. The findings and conclusions expressed are solely those of the authors and do not reflect the opinions or policy of our employers, the organizations that supported this work, the Federal Reserve Bank of New York, or the Federal Reserve System.

The Quarterly Journal of Economics (2026), 1825–1870. https://doi.org/10.1093/qje/qjag027.
Advance Access publication on May 12, 2026.

relative to older graduates' post-pandemic (2022–2024) compared to pre-pandemic (2017–2019), a pattern we do not observe in non-remotable jobs. *JEL codes:* J24, M15, M53, M54, J16, O33, R23.

I. Introduction

Coworkers are more distant than ever before. In 2024, workers lived an average of 27 miles from the office, nearly three times farther than in 2019 (Akan et al. 2024). Remote and hybrid arrangements have become common: employees work from home one to two days a week on average (Flood et al. 2024), and even on days when they are in the office, many of their colleagues are still at home or in distant offices (Goldberg 2021). The growing distance may make it harder to learn from colleagues, who traditionally have been the source of about a sixth of lifetime human capital (Herkenhoff et al. 2024). Yet because work increasingly takes place on screens (Belden 2022), it is possible that workers can learn just as effectively from afar. In an increasingly digital world, how does proximity to coworkers affect workers' productivity and on-the-job training?

To answer this question, we study software engineers at a Fortune 500 online retailer from 2019 to 2024. We focus on engineers based in the headquarters campus whose teammates varied in their physical proximity. Before the pandemic, some teams were co-located in a single building; other teams were split across two headquarters buildings, roughly a 10-minute walk apart; still others were completely distributed, with teammates working remotely or on satellite campuses. When the offices closed for the pandemic, these differences became immaterial. By comparing differences across team types before the offices closed—when they varied in proximity—to afterward—when they did not—we can identify the causal effects of proximity on engineers' training and productivity. We can further investigate whether these differentials re-emerge after the firm's return-to-office (RTO) mandates in 2022 and 2023, which brought co-located teams—those assigned to a single building—back together, while leaving geographically distributed teams apart.

We find that sitting alongside teammates increases the digital feedback engineers receive on their code. Before finalizing their code, engineers must have it reviewed by colleagues who flag problems and suggest improvements. When the offices were open, engineers on co-located teams received 23.9% more comments (1.92 more per program) than engineers on multi-building teams—those split across the firm's two headquarters buildings. Once the offices closed, this advantage largely disappeared, narrowing by 18.3% (1.47 comments per program, $p =$ .0026). The decline stemmed entirely from reduced feedback from teammates, rather than from other colleagues. Using a machine-learning algorithm to classify comment quality, we find that the lost comments were disproportionately those predicted to be helpful, actionable, impactful, and well reasoned.

Proximity makes it easier to ask for feedback. Engineers sitting near their teammates ask more follow-up questions, consistent with neuroscience studies showing that in-person conversation generates stronger feelings of closeness than video calls and instant messages (e.g., Seltzer et al. 2012; de Groot et al. 2015; Schwartz et al. 2022). When proximate to their teammates, engineers also tap a wider range of people for feedback. This result mirrors classic studies showing that in-person interaction facilitates, rather than replaces, phone and email communication (Gaspar and Glaeser 1998; Allen and Henn 2007; Agrawal and Goldfarb 2008) and echoes recent research showing that remote work silos communication networks (Yang et al. 2022; Gibbs, Mengel, and Siemroth 2023). Even with modern technology, face-to-face interaction is a complement rather than a substitute for digital communication.

Proximity is fragile. Even a 10-minute walk between buildings on the same campus reduces feedback as much as being multiple states away, suggesting that small frictions to face-to-face contact can have outsized effects. In addition, a single distant colleague can generate negative externalities for the rest of the team: when a new hire turns a co-located team into a distributed one, feedback between the original teammates drops sharply, even though they still sit together. In contrast, adding a new co-located teammate has no such effect. Moving team meetings online to accommodate a distant colleague appears to weaken connections among those who remain together.

Proximity does not affect all engineers equally; instead, it most affects those with the most to learn from their colleagues. New hires and young engineers receive more feedback when co-located and lose more mentorship when remote. Female engineers are also more affected by proximity, partly because they ask fewer coworkers for feedback when they cannot do so in person.

Mentoring improves engineers' code quality in the short and long run. Around the RTO mandates, engineers on co-located teams, who saw larger increases in face-to-face time with teammates, experienced greater improvements in code quality than those on geographically distributed teams. After the three-days-per-week office mandate was introduced, engineers on co-located teams became 2.2 percentage points less likely to add files that were subsequently deleted ($p = .041$) and 1.4 percentage points less likely to introduce bugs ($p = .022$) than engineers on distributed teams. Less-tenured and younger engineers experienced roughly twice the gains of the average engineer. The benefits of co-location are persistent: engineers with more experience on co-located teams continued to write better code, even after differences in proximity and feedback disappear.

However, there is a cost to mentorship: providing feedback and improving others' code requires effort. Experienced engineers, who offer the most feedback, write less code when sitting near their colleagues. Pre-pandemic, senior engineers on co-located teams wrote less code than their counterparts on distributed teams. When the offices closed, this gap closed; when the offices reopened, it widened again. The firm faces a trade-off between boosting junior engineers' skills and allowing senior engineers more time to program.

When mentorship falters and talent is harder to build in-house, firms can instead buy it by hiring more experienced workers. We find evidence of this response: the firm shifted toward hiring older workers when the offices closed and then back toward younger hires once they reopened. National data suggest that this firm is not unusual: in remotable jobs, young workers have seen larger increases in unemployment between 2017–2019 and 2022–2024 than did older workers, following the persistent rise in remote work. No comparable gap emerges in non-remotable jobs, a difference that emerged before the spread of generative AI and persists after accounting for occupational exposure to AI (Eloundou et al. 2024; Schubert 2025). The labor market pressures on young people in remotable jobs help explain broader trends. Our estimates suggest that remote work accounts for 64% of the total unemployment increase among young college graduates over this period.

We also see evidence of the desire to buy talent by looking at who is poached from the firm we study. When the offices were closed, engineers trained on co-located teams were more likely to leave for better jobs, suggesting that investing more in workers made it more likely they would leave the firm. Because the firm does not fully capture the returns to these investments, it might underincentivize coworkers' investments in each other's human capital, even if it could perfectly observe them. This classic problem in general human capital formation (Becker 1964) makes coworkers' social bonds to one another—which are often forged face to face—critical in facilitating investment.

Consistent with having the most to gain from proximity, young and less-tenured engineers come into the office more during the RTO periods, as evidenced by the firm's badge data. They are particularly likely to go in when their teammates are based in the same office, suggesting that the prospect of being near colleagues is a key draw for those with the most to learn. This pattern holds nationally: in 2022–2024, younger workers were more likely to be back in the office, both in tech and among college-educated workers broadly, even when limiting to nonparents. Yet the office only facilitates mentorship when colleagues are present; if teammates remain remote, young workers cannot easily sidestep the scarring effects of distance on their human capital development.

We provide evidence that proximity to coworkers increases on-the-job training and helps young and less-tenured workers build human capital. This finding highlights several key dimensions of heterogeneity in the effects of remote work. Remote work may harm future productivity while boosting current productivity; it may harm the quality of work while increasing its quantity; and it may disadvantage inexperienced workers while helping experienced workers to thrive. These heterogeneous effects can help reconcile the wide range of productivity estimates in the literature, which tend to find positive effects on the quantity of work in settings with experienced workers (Bloom et al. 2015; Choudhury, Foroughi, and Larson 2021; Bloom, Han, and Liang 2024; Angelici and Profeta 2024; Choudhury et al. forthcoming; Fenizia and Kirchmaier 2024), while finding less positive or even negative effects on work quality especially when workers are less experienced (Dutcher 2012; Battiston, Blanes i Vidal, and Kirchmaier 2021; Atkin, Chen, and Popov 2022; Gibbs, Mengel, and Siemroth 2023; Emanuel and Harrington 2024; Ho, Jalota, and Karandikar 2026; Jalota and Ho 2024; Wang, Li, and Yu 2025). These dimensions of heterogeneity are not the only drivers of differences across studies—remote work, for instance, appears consistently more challenging in developing countries (Atkin, Chen, and Popov 2022; Ho, Jalota, and Karandikar 2026; Jalota and Ho 2024). But accounting for worker learning is essential to understanding the full range of remote work's effects.

Our findings also help resolve the puzzle of remote work's rarity before the pandemic (Mas and Pallais 2020), which seemed at odds with workers' strong preference for remote work (Mas and Pallais 2017; He, Neumark, and Weng 2021; Lewandowski, Lipowska, and Smoter 2026; Maestas et al. 2023; Cullen, Pakzad-Hurson, and Perez-Truglia 2025) and its often positive short-run effects on output. One explanation is that the value of the office lies in training for tomorrow and improving the quality, not the quantity, of work today.

This article also speaks to the debate over how changes in the nature of work affect workers at different stages of their careers. We find that remote work depresses demand for junior versus senior talent, aligning with Schubert (2025)'s finding that remote work raises experience requirements in job postings. The resulting increase in intergenerational inequality may compound those of generative AI, which recent evidence also finds reduces demand for junior talent (Brynjolfsson, Chandar, and Chen 2025; Hosseini Maasoum and Lichtinger 2025).

The rest of the article is organized as follows. Section II provides context on software engineering. Section III introduces the data. Section IV describes the empirical design. Sections V, VI, and VII discuss how proximity impacts feedback, code quality, and coding output, respectively. Sections VIII, IX, and X consider implications for office attendance and hiring in the firm and nationally. Section XI concludes.

II. Background on Software Development

Our data come from a Fortune 500 online retailer, whose software engineers perform tasks typical of the industry: building the website's front-end interface, maintaining the back-end product catalog, developing internal tools for other parts of the firm (e.g., the finance team), and working on AI-driven features like product recommendation services.

The engineers follow a standard process for developing business-critical code on GitHub. When making changes, an engineer creates a "branch" from the primary codebase, which separates their rough drafts from live code. Before their changes can be merged into live code, they must be reviewed. This key quality control step lets other engineers verify that the new code is well tested, error free, and understandable to future engineers (including the author's future self). Typically, two reviewers weigh in: one from the author's immediate team and one with relevant external experience, such as familiarity with that part of the codebase. Strong norms and managerial expectations encourage engineers to give feedback when asked, but there are no explicit incentives to comment extensively, since such incentives would be too easily gamed with superficial comments.

In addition to vetting the current code, reviewers can build the author's skills. One manager explained, "We ask senior, technical folks . . . to make their code reviews a learning opportunity by, for example, including the reasoning behind suggested changes." In line with this goal, the reviewer has more experience at the firm than the code's author in 70% of reviews.

Teams follow an Agile management system that includes daily stand-up meetings where everyone literally stands to keep meetings brief. In these 10–15-minute meetings, engineers share progress updates and flag anything blocking their work. These meetings also give engineers a chance to ask others to review their code, a request which many prefer to make in person rather than digitally (e.g., on Slack or GitHub).

The format of these daily meetings depends on teammates' physical proximity. Most teams follow an implicit "one Zoom, all Zoom" norm: if one person cannot be physically present, everyone pulls out their laptops and gets on a video call. As one engineer noted, "[my team] would almost never book a room and held all of our meetings [online] since we had a remote team member." Even in the headquarters campus, a short walk between buildings was enough to push some meetings online. It was hard to justify a 10-minute walk each way for an equally short meeting. Beyond daily stand-ups and the chitchat before and after them, co-located engineers also had more chances for informal face-to-face interactions around their desks due to the firm's open office plan. We investigate the ramifications of this close proximity on engineers' training and productivity.

III. Data

We collate data to (i) characterize workers' backgrounds, (ii) identify workers' physical proximity to their teammates, and (iii) measure different facets of workers' productivity and on-the-job training. Together, our different data sources span from mid-2019 to mid-2024, covering the office closures of 2020 and two RTO waves in 2022 and 2023.

III.A. Workers' Backgrounds and Demographics

Many of the firm's engineers are young and new to the company. Before the closures, engineers were, on average, 29 years old with just 1.4 years at the company, suggesting many were still building both general and firm-specific human capital (Table I, first two rows). As is common in software engineering, the workforce is male-dominated: men compose 81% of our sample and 75% of programmers nationally (Ruggles et al. 2022). Only 22% of engineers had children—and just 17% of female engineers—consistent with the workforce's relative youth.[1]

III.B. Identifying Teammates' Physical Proximity

We use personnel data to distinguish between co-located and distributed teams. We define an engineer's teammates as all the employees reporting to the same manager.[2] Using each teammate's assigned office, we then classify teams as (i) co-located in one building, (ii) split across multiple buildings on the firm's headquarters campus (a 10-minute walk apart), or (iii) geographically distributed, with some members working remotely or in satellite offices. Throughout, we focus on engineers in the headquarters campus, whose proximity to teammates varies. Before the pandemic, 49% of headquarter-campus engineers were on co-located teams, 34% were on teams split across buildings, and 17% were on geographically distributed teams.

By the time the offices reopened, the firm had consolidated its headquarters operations into one building, while expanding its satellite offices and remote workforce. Consequently, only 25% of headquarter-campus engineers were on co-located teams, with the rest geographically distributed.

1. Badge Data Around RTOs. From 2022 onward, we observe office attendance using badge data, which record each employee's daily entry into the firm's buildings. During the first RTO—which encouraged two in-office days a week—engineers came in an average of 0.6 days a week. During the second

1. Starting in June 2020, the firm began collecting caregiving information, including childcare responsibilities. These data cover 70% of software engineers in our sample.

2. This strategy identifies the teammates for the plurality of workers. We exclude the 1.2% of engineers with high-level managers who oversee multiple teams, but the results are similar without this restriction.

TABLE I
SUMMARY STATISTICS: CO-LOCATED AND MULTI-BUILDING TEAMS

	Mean	P10	P50	P90	Mean: Co-located	Mean: Multi-building	Difference	
Age (years)	28.76	23.00	28.00	35.00	28.51	29.09	−0.58	0.36
							(0.42)	(0.56)
Firm tenure (years)	1.36	0.17	0.92	3.50	1.21	1.56	−0.34***	−0.42***
							(0.11)	(0.12)
% female	18.58	0.00	0.00	100.00	19.53	17.33	2.19	−2.75
							(2.78)	(3.26)
Job traits								
Job level	1.71	1.00	2.00	3.00	1.62	1.82	−0.20***	−0.06
							(0.06)	(0.07)
Salary + stocks	121,262	100,000	115,000	150,050	119,075	124,161	−5,086***	−894
							(1,810)	(2,210)
Team traits								
# teammates	6.09	3.00	6.00	10.00	5.72	6.57	−0.85**	−0.42
							(0.42)	(0.47)
Manager tenure	2.87	0.78	2.33	5.90	2.84	2.92	−0.08	−0.41
							(0.32)	(0.36)
Manager job level	3.30	2.00	3.00	4.00	3.21	3.42	−0.21**	−0.08
							(0.09)	(0.10)

TABLE I
CONTINUED

	Mean	P10	P50	P90	Mean: Co-located	Mean: Multi-building	Difference	
Engineer group								
Back end	12.73	0.00	0.00	100.00	21.23	1.48	19.75***	—
							(3.46)	
Front end	19.83	0.00	0.00	100.00	30.47	5.76	24.71***	—
							(4.61)	
Internal tools	60.00	0.00	100.00	100.00	37.23	90.13	–52.90***	—
							(5.26)	
AI features	7.44	0.00	0.00	0.00	11.07	2.63	8.44***	—
							(3.13)	
Engineer group controls								✓
# software engineers	1,055	1,055	1,055	1,055	637	418		
# teams	304	304	304	304	206	121		

Notes. This table shows traits of the engineers, their jobs, and their teams before the offices closed for COVID-19. The sample includes engineers whose teams are all in the headquarters campus. P10 refers to the 10th percentile, P50 to the median, and P90 to the ninetieth percentile. The last two columns present the pre-closure difference between engineers on co-located and multi-building teams. The last column includes engineering group controls: indicators for whether the engineer works on front-end website design, back-end catalog management, internal tools for others in the company, or AI features. "Job level" refers to the engineer's position within the firm's hierarchy from zero (an intern) to six (senior staff). Standard errors in parentheses are clustered by engineering team. * $p < .1$; ** $p < .05$; *** $p < .01$.

RTO—which called for three in-office days a week—office attendance rose to 1.9 days a week. Throughout both periods, the firm did not specify which days employees should come in, but most engineers came in on Tuesday through Thursday (Online Appendix Figure A1).

III.C. Measuring Productivity and Learning

We have two extracts of data from the firm's code development system covering different periods. Around the pandemic office closures, we observe the quantity of code engineers wrote in the firm's main codebase and the feedback they received on it. Around the RTOs, we observe code quantity across all codebases and proxies for its quality.

1. Data Around the Office Closures. Our first extract spans from August 2019 until December 2020 and tracks all changes to the main codebase along with the feedback that engineers received on this code. In our analysis sample, 1,055 engineers made 29,809 distinct changes to the codebase and received 174,014 comments from their coworkers.

For every distinct change, we see who made the change, when it occurred, how many lines of code were added, and how many files in the main codebase were changed. The average modification—or "program"—adds 346 lines across seven files. Engineers typically write two programs in the main codebase each month (see Online Appendix Table A.1 for the distribution). Our preferred metric of immediate productivity is programs written, as engineers are encouraged to write shorter, more modular programs that are easier to test and debug.

The data also detail every comment exchanged during each code review, including its text, timestamp, and the identity of the commenter. These comments function much like feedback on a paper draft: the commenter highlights specific portions of the code and comments on its structure, functionality, or readability. Engineers receive an average of 6.5 comments on each program (Online Appendix Table A.1 shows the distribution). Each comment averages 16 words (100 characters). The feedback is usually substantive, not only telling the engineer how to change the code but also explaining the underlying reasoning (see Section V.C). Thus, this extract lets us measure investments in engineers' human capital.

2. Data Around the RTOs. The second extract measures code quantity and quality for all the firm's codebases from December 2020 to mid-2024. As before, these data track how frequently engineers make changes and the extent of those changes, but now spanning all codebases rather than only the main one.

This extract includes two code-quality metrics. Engineers often assess quality along two dimensions: churn—whether engineers are ineffectively spinning their wheels—and bugs—whether they introduce errors or vulnerabilities. We measure churn as the frequency with which engineers add files that are deleted in the next six months, which occurs in 15% of programs. A file may be deleted because it was a tangled mess of logic (also known as "spaghetti code") or because it introduced a feature the firm later abandoned. Either way, such disposable code is not a good sign of the quality or utility of an engineer's code. We measure bugs by identifying changes that are immediately and fully reverted, typically indicating that the engineer's changes precipitated an emergency requiring a rapid rollback to an earlier version of the code. These more serious problems occur in 3.5% of programs.

IV. Empirical Design

Our goal is to identify how proximity to teammates affects engineers' investments in each other's human capital, as well as the quantity and quality of their code. We exploit two key shocks to proximity: the office closures of COVID-19 and the later RTO mandates. In both cases, co-located teams saw bigger changes in proximity than distributed ones, facilitating difference-in-differences (DiD) designs.

IV.A. Office Closure Design

When the offices closed, engineers on co-located teams saw larger losses in proximity to their teammates than engineers on distributed teams. We leverage this difference in the following DiD design:

$$\begin{aligned} \mathrm{Y}_{it} = {} & \beta \text{Post Closure}_t \cdot \text{Co-Located Team}_i + \mu_i + \mu_t \\ & + X_{it}'\Gamma + \epsilon_{it}, \end{aligned} \tag{1}$$

where Y_{it} represents either the feedback that engineer i received on her programs in month t (in Section V) or the number of programs she wrote (in Section VII), but not the quality of her code, data on which are only available after the closures. The specification includes fixed effects for engineer (μ_i) and month (μ_t). Because this design considers a single focal event, it does not run into the problems that can arise with staggered treatment timing (e.g., Goodman-Bacon 2021). We cluster standard errors by team, the unit of treatment assignment. We define Co-Located Team$_i = 1$ if an engineer was consistently on a co-located team—with all members assigned to the same building—throughout the pre-period, ensuring a stable treatment definition. The results are similar when we define team type contemporaneously.

Our primary identification strategy focuses on engineers whose teammates were all on the headquarters campus, but whose team was either all in one building or split across two. In this sample, co-location often came down to chance: when a new hire joined, was there an open seat near the rest of the team? These seating logistics were more challenging for engineers working on internal tools, who aimed to sit near both end users and fellow engineers, resulting in lower co-location rates (35%). Engineers on co-located and multi-building teams are broadly similar in their observable characteristics (Table I, seventh column). Accounting for engineering group, engineers on one- and multi-building teams look observably similar on all dimensions except tenure (Table I, last column).[3]

For β to reflect the causal effect of losing proximity, a parallel-trends assumption must hold: absent the closures, feedback to engineers on co-located and multi-building teams would have evolved similarly. Reassuringly, pre-closure trends in feedback are parallel (Figure I). Nevertheless, potential threats remain. Workers and managers may sort onto co-located teams, and tasks may differ across team types, differences that could interact with time-varying shocks, particularly those from the pandemic.

Our preferred specification aims to relax this identifying assumption by including month-specific controls for the engineer's group, tenure, and age. These controls absorb residual variation, address imbalances across co-located and multi-building teams, and absorb interactions between these factors and the pandemic. For instance, if the pandemic affected website engineers differently from those working on internal tools, that difference would

3. In supplementary analysis, we compare co-located and geographically distributed teams, but geographically distributed engineers are more selected: older, more tenured, and higher level (Online Appendix Table A.2).

be captured by month-by-group fixed effects. Likewise, if the pandemic posed particular challenges for engineers who were new to the firm, that difference would be absorbed by month-by-tenure fixed effects. When analyzing feedback, we also control for program scope (quartics in the number of lines added, lines deleted, and files changed), which may mediate the extent of feedback engineers receive. Our full set of controls also includes month-specific controls for team size, gender, race, home ZIP code, job level, and initial building assignment. The robustness of our results to the inclusion of various controls lends confidence to the design.

As a placebo check, we examine feedback from non-teammates, who experienced no differential change in proximity across co-located and multi-building teams. The null result for this check helps assuage identification concerns and contradicts some alternative explanations. If engineers on co-located teams simply needed more feedback or undertook projects requiring more input before the closures but not afterward, we would expect the effects to show up in non-teammate feedback as well. Instead, the differential changes are confined to teammate feedback—where proximity changed differentially—and are absent for non-teammate feedback, where proximity evolved similarly for all engineers.

Finally, as a complementary strategy, we leverage within-engineer switches in team type. Before the office closures, 3.7% of engineers switched between co-located and multi-building teams, and another 6.0% saw changes in teammate proximity because of colleagues joining, leaving, or switching teams. These switches are associated with changes in feedback when the offices are open, but not after they close. This pattern holds when including engineer and engineer-by-manager fixed effects, helping address concerns that selection into team type—at the worker or manager level—drives the results.

IV.B. Office Reopening Design

During the RTO period, the firm's headquarters campus had a single building, so we compare engineers on co-located (same-city) teams to those on geographically distributed teams, defined as having at least one member at a different campus or working permanently remotely.[4] When the offices reopened, engineers on

4. Permanently remote workers were given exceptions typically because they lived far from any campus, averaging 123 miles from the closest office, compared with 25 miles for on-site engineers.

co-located teams saw larger gains in proximity than those on distributed teams. We estimate:

$$Y_{it} = \sum_{\tau \in \{\text{Closed}, \text{1st RTO}, \text{2nd RTO}\}} \gamma_\tau \text{ Co-Located Team}_{it} \ \mathbb{1}[t \in \tau]$$

$$+ \mu_i + \mu_t + X_{it}'\Theta + u_{it}, \tag{2}$$

where Y_{it} is either the number of programs written or a measure of program quality, but not feedback received, data on which are only available until December 2020. This specification allows the effect of being on a co-located team to vary across the office closures, the first two-day-a-week RTO, and the second three-day-a-week RTO. We allow Co-Located Team$_{it}$ to vary over time because, over this multiyear period, nearly half of engineers (48.6%) spent time on both co-located (same-city) teams and geographically distributed teams. The resulting within-engineer variation—with hundreds of switches per period—allows us to identify differences across team types in every period even with individual fixed effects (μ_i). Throughout, we focus on engineers who worked on the headquarters campus but whose teammates' locations varied.

The identifying assumption is that switching across team types is uncorrelated with other shocks to an engineer's code quantity or quality. One concern is that assignment to co-located versus geographically distributed teams appears less random than assignment to co-located versus multi-building teams: engineers on geographically distributed teams tend to be older, more tenured, and more likely to work on internal tools (Online Appendix Table A.3). Our preferred specification addresses these differences by including month-by-tenure, month-by-age, and month-by-engineering-group fixed effects. We also show robustness to the full suite of controls.

Our strongest test of the identifying assumption exploits the fact that switching team types should only affect coding when it changes physical proximity to teammates. We examine three distinct periods—the office closures, the first RTO (two days a week), and the second RTO (three days a week)—and generate testable predictions for each. We predict that $\gamma_{closed} \approx 0$ because being assigned to the same office as one's teammates is inconsequential when that office is closed. We further expect that $\gamma_{2nd\,RTO} > \gamma_{1st\,RTO}$, since the second RTO entailed more days in the office and more overlap in teammates' schedules, since Tuesday–Thursday all became heavily trafficked days (Online Appendix Figure A.1).

If switches across team types were simply correlated with unrelated shocks to code quantity or quality, this specific pattern of temporal heterogeneity would be difficult to explain.

V. Proximity and Feedback

If physical proximity causally boosts online feedback, three patterns should emerge around the office closures: (i) co-located engineers should receive more feedback when the offices are open, (ii) feedback should decline for everyone once the offices close, and (iii) the gap in feedback between previously co-located and already distributed teams should narrow. All three predictions are borne out in Figure I, Panel A. While the offices were open (left of the vertical line), engineers on co-located teams received 23.9% (or 1.92) more comments on their programs than engineers on multi-building teams, even after controlling for program length, engineer tenure, age, and engineering group ($p = .0005$). Once all engineers were remote (right of the vertical line), this difference shrank to 7.1% (or 0.57 comments per program), with the sharpest decline coinciding with the closures.

Our DiD design, comparing the pre- and post-closure feedback gap across team types, indicates that the greater loss of proximity among co-located teams reduced the feedback they received by 16.8%. Including individual engineer fixed effects, the DiD coefficient is 18.3% (or 1.47 comments per program, $p = .0026$, Table II, column (3)). This estimate remains similar across alternative specifications, ranging from no controls ($R^2 = 1.5\%$) to the full set ($R^2 = 61\%$, Table II).[5] Results are similar when measuring feedback as total characters or words (Table III, Panel A, columns (1) and (2)), and effects are even larger when focusing on substantive, technical feedback, as discussed in Section V.C. Furthermore, when teammates are proximate, they give feedback more quickly (Table III, Panel A, column (3)).

Notably, the differences in feedback between engineers on co-located and multi-building teams are driven solely by feedback from teammates, with no detectable effect on feedback from engineers on different teams (Table II, columns (5) and (6)). This null contradicts some alternative explanations. If engineers on co-located teams simply needed more feedback, we would expect this

5. Our full set of controls also include team size, demographics, ZIP code, job level, and initial building.

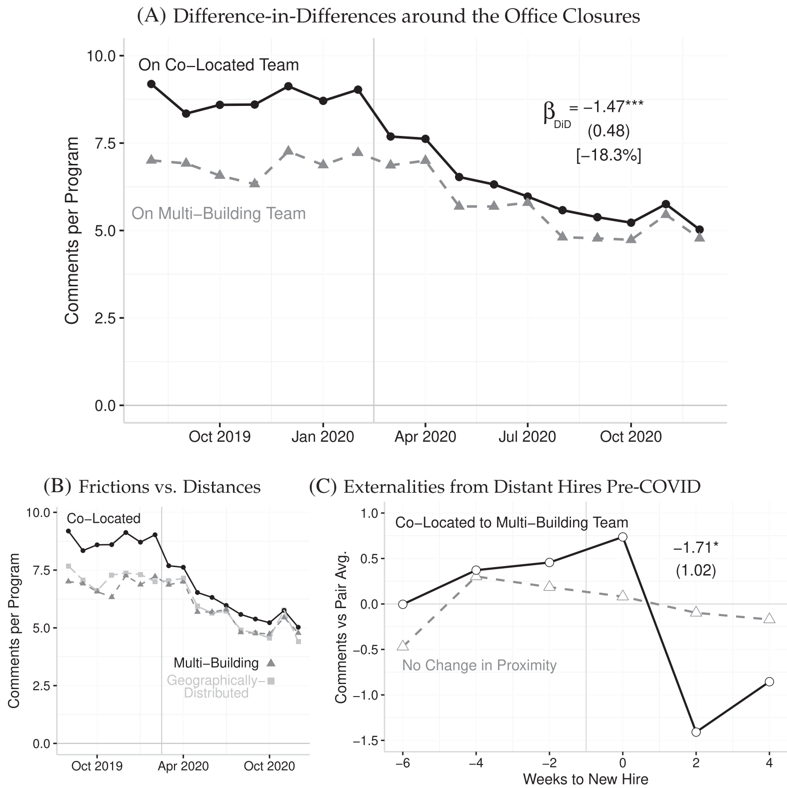

FIGURE I

Proximity to Teammates and Online Feedback

This figure analyzes whether engineers who are physically proximate to their teammates receive more feedback online. Panel A illustrates our difference-in-differences (DiD) design that compares engineers on co-located teams ($N = 637$) to engineers on multi-building teams ($N = 418$) around the COVID-19 office closures (gray line). The plot residualizes by our preferred controls for program scope, engineer age, tenure, and engineering group. The reported coefficient is the pooled DiD coefficient with engineer fixed effects (Table II, column (6)), with the percent change in brackets. Panel B extends the analysis to include teams geographically distributed across campuses. Panel C shows a complementary, pre-COVID design, which compares new hires that convert teams from co-located to multi-building teams ($N = 16$ teams) versus new hires that do not change the co-location of their team ($N = 119$ teams). This analysis is limited to comments between pairs of co-located teammates who both predate the six-week pre-period. The plot shows comments per program relative to the average in the coder-commenter pair, with the annotated DiD coefficient from equation (3). Standard errors are clustered by team. * $p < .1$; ** $p < .05$; *** $p < .01$.

TABLE II

PROXIMITY TO TEAMMATES AND ONLINE FEEDBACK

	Comments per program					
					Teammate	Non-teammate
	(1)	(2)	(3)	(4)	(5)	(6)
Post × co-located	−1.29***	−1.35***	−1.47***	−1.25**	−1.33***	0.12
	(0.48)	(0.49)	(0.48)	(0.49)	(0.38)	(0.31)
Co-located team	1.16**	1.92***				
	(0.52)	(0.55)				
Post	−1.22***					
	(0.36)					
Pre-mean, co-located	8.04	8.04	8.04	8.04	4.28	3.73
Post × co-located as %	−16.1%	−16.8%	−18.3%	−15.6%	−31%	3.3%
Preferred controls		✓	✓	✓	✓	✓
Engineer fixed effects			✓	✓	✓	✓
Full controls				✓	✓	✓
# engineer-months	9,304	9,304	9,304	9,304	9,304	9,304
R^2	0.02	0.43	0.56	0.61	0.54	0.63

Notes. This table examines the relationship between physical proximity to teammates and the online feedback that engineers receive on their code, using the difference-in-differences model in equation (1). Columns (5) and (6) present a placebo check: proximity to teammates should impact feedback from teammates but not from non-teammates. Preferred controls include engineer age, tenure, and group, all interacted with month, plus program scope (quartics of lines added, lines deleted, and files changed). Full controls add gender, race, home ZIP code, job level, and initial building, also interacted with month. Standard errors (in parentheses) are clustered by team. * $p < .1$; ** $p < .05$; *** $p < .01$.

TABLE III

PROXIMITY AND OTHER DIMENSIONS OF ONLINE FEEDBACK

	(1)	(2)	(3)	(4)
Panel A: Extensiveness and timeliness	Total characters	Total words	Hours to comment	% Slack
Post × co-located	−164.10**	−26.87**	1.12*	−1.58*
	(67.78)	(11.09)	(0.64)	(0.95)
Pre-mean, co-located	833.24	136.92	16.02	4.06
Post × co-located as %	−19.7%	−19.6%	7%	−38.9%
Panel B: Intensive and extensive margins	Comments per commenter	Commenters per program	Distinct commenters per program	
Post × co-located	−0.63***	−0.05	−0.12**	
	(0.23)	(0.06)	(0.06)	
Pre-mean, co-located	4.36	1.77	1.37	
Post × co-located as %	−14.3%	−2.6%	−9%	
Panel C: Back-and-forth conversations	Commenter's	Program writer's		Commenter's
	initial comments	Replies	Questions	follow-ups
Post × co-located	−0.71**	−0.72**	−0.15***	−0.76**
	(0.28)	(0.29)	(0.06)	(0.37)
Pre-mean, co-located	4.91	2.14	0.24	3.13
Post × co-located as %	−14.4%	−33.7%	−62.7%	−24.3%

Notes. This table considers alternative metrics of online feedback. In Panel A, Column (4) is the percent of reviews with references to Slack discussions. In Panel B, if a programmer wrote five programs in a month and had the same person comment on all of them, she would have one *Commenters per program* but only 0.2 *Distinct commenters per program*. *Distinct commenters per program* aims to approximate the size of the engineer's network. Each specification replicates Table II, column (3). * $p < .1$; ** $p < .05$; *** $p < .01$.

to show up in non-teammate feedback as well. Similarly, if the projects undertaken by co-located teams happened to necessitate more feedback before the closures but not afterward, the resulting decline in feedback would have occurred across all sources, not just in their own small teams.

Turning to engineers who switch team types, we find that the same engineer receives more feedback when on a co-located team than when on a multi-building team—but only when the offices are open (Online Appendix Table A.4, column (1)). Once offices close and physical assignments become irrelevant, the co-location advantage disappears. Suggestively, these patterns persist when we include worker-by-manager fixed effects, indicating that the same engineer working under the same manager receives more feedback when her team is co-located than when it becomes distributed (e.g., because a new hire cannot find a seat near the rest of the team, Online Appendix Table A.4, columns (3) and (4)).[6]

Face-to-face and digital communication appear to be complements rather than substitutes: instead of compensating for lost in-person interaction with more online feedback, engineers who lose physical proximity exchange less online feedback as well. We find the same pattern in references to other communication channels (e.g., Slack) in code reviews (Table III, Panel A, column (4)). To the extent that proximate teammates also communicate more in person, our estimate of proximity's effect on online mentorship represents a lower bound on its total effect.

1. Drivers of the Complementarity. Being face to face with teammates appears to make engineers more comfortable seeking online feedback—both from a wider range of colleagues (the extensive margin) and in greater depth (the intensive margin). On the extensive margin, proximity expands engineers' feedback networks (Table III, Panel B, column (3)). Engineers on co-located teams are less likely to return repeatedly to the same commenter,

6. We consider a number of additional robustness checks. First, while our main analysis focuses on headquarters-campus engineers (to ensure more apples-to-apples comparisons), our findings are similar if we instead analyze all engineers regardless of their location (Online Appendix Figure A.2, Panel B). Second, our results are also similar if we limit our analysis to internal tool engineers, who are more likely to be distributed to work near the users of their tools (Online Appendix Figure A.2, Panel C). Finally, our results are comparable if we consider teams that were more or less co-located than average rather than whether all teammates were co-located (Online Appendix Figure A.2, Panel D).

instead drawing on a wider pool of commenters when the offices are open.

On the intensive margin, losing proximity reduces all aspects of the conversations about the code (Table III). Much of this effect operates through follow-up questions and replies: engineers on co-located teams ask 48.4% (0.12) more follow-up questions when the offices are open, a differential that vanishes once they close (Table III, Panel C, column (3), $p = .0083$). Roughly half of proximity's total effect on feedback comes through these follow-up exchanges, with the other half coming from initial reviewer feedback (column (1)).

V.A. What Features of Proximity Matter?

1. Small Frictions. Small barriers to face-to-face contact can undermine feedback as much as large ones. Figure I, Panel B shows that engineers on teams separated by a 10-minute walk receive no more feedback than those on teams spread across the country—both of whom receive less feedback than engineers on co-located teams. These findings accord with research on academics showing that being in a different building, or even on a different floor, reduces coauthorship (Kraut, Egido, and Galegher 1988; Catalini 2018; Salazar Miranda and Claudel 2021). Our results show that these small frictions also have big effects for teammates who share projects and meet daily.

2. Externalities from Distant Teammates. The firm's "one Zoom, all Zoom" norm means that a single distant teammate can shift the entire team to virtual interaction. Consistent with this norm, we find that having just one distant teammate reduces feedback even among teammates who sit together. While the offices are open, engineers in the same building exchange 14.5% less feedback when even one teammate is in another building, relative to pairs whose whole team is in the same building ($p = .037$, Online Appendix Figure A.3).

New hires provide further evidence of these externalities. When no desks are available near the rest of the team, a new hire can transform a co-located team into a multi-building one. We find the transition to a multi-building team is associated with a sharp decline in the feedback exchanged between the original teammates who continue to sit together (the solid line in Figure I, Panel C). By contrast, adding a new hire who does not disrupt the team's co-location produces no such change (the dashed line in Figure I, Panel C). Taking the difference in these differences implies that gaining a single distant teammate reduces feedback by 1.71 comments per program ($p = .095$).[7] This decline is large enough to meaningfully erode the benefits of proximity for those who remain co-located.

V.B. Substantive Feedback

During code reviews, engineers exchange substantive feedback about their code, which diminishes when engineers are no longer co-located. Figure II, Panel A tracks how losing proximity to teammates affects the frequency of the 100 most common words in the comments, excluding stop words like pronouns and prepositions (Bird, Klein, and Loper 2009). When engineers lose proximity to their teammates, the frequency of almost all of these words declines. Notably, many of these words are directly tied to how the code works—for example, about its functions, methods, models, and return values—and what the programmer has done to check programs to ensure they do not throw exceptions or create other bugs. These results suggest that losing proximity does not merely trim the fat of nitpicky comments but instead reduces substantive feedback that may be critical for improving code quality.

Similar patterns emerge when we use supervised machine learning to assess comment quality. We hired external software engineers to label a random sample of 5,377 comments as (i) helpful, (ii) explaining the underlying reasoning for suggested changes, (iii) actionable, and (iv) likely to change the code (see Online Appendix III.A for details). Evaluators tended to view the comments positively, judging 76% to be helpful, 51% as explaining their reasoning, 60% to be actionable, and 52% as likely to change the code. We trained a gradient-boosted decision tree (Chen and Guestrin 2016) on the text of these labeled comments to predict the ratings for all 174,014 comments in the data (see Online

7. Sixteen teams (with 46 engineers) switch from co-located to multi-building teams around a new hire, and 117 teams (with 369 engineers) hire someone new but do not change their co-location. Our DiD estimates

$$\frac{\text{Comments}}{\text{Review}}_{ijt} = \alpha \text{Post Hire}_t \times \text{From One to Multi-Building Team}_{ij} + \psi \text{Post Hire}_t + \mu_{ij} + \epsilon_{ijt}, \tag{3}$$

where μ_{ij} represents programmer (i) by commenter (j) pair fixed effects, ψ captures the impact of any new hire, and α captures the additional effect of a new hire who makes the team distributed.

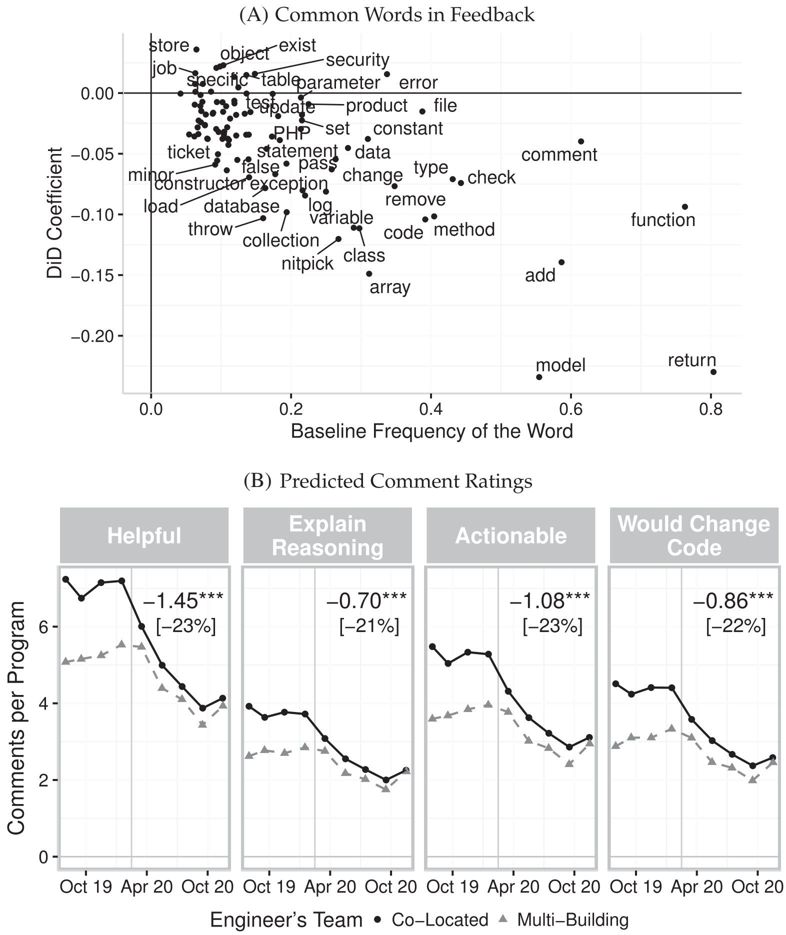

FIGURE II

Effects of Losing Proximity on Substantive Feedback

This figure examines feedback content. Panel A illustrates how losing proximity to teammates affects individual word usage, with each word's pre-closure frequency on the x-axis and the difference-in-differences (DiD) coefficient on the y-axis. Panel B shows the impact of losing proximity on predicted comment quality. Predictions come from a supervised machine-learning algorithm applied to the human ratings of external software engineers (see Online Appendix C). The annotated coefficient reflects the DiD estimate, with percentage effects in brackets. All DiD specifications include engineer fixed effects and our preferred month-specific controls for engineer age, tenure, and engineering group. Standard errors are clustered by team. * $p < .1$; ** $p < .05$; *** $p < .01$.

Appendix III.B for details), achieving 64%–78% accuracy in hold-out samples.

Figure II, Panel B shows that losing proximity to teammates reduces comments that would likely be rated positively. Before the office closures, engineers on co-located teams received more comments predicted to be helpful, well reasoned, actionable, and likely to change the code. Once the offices closed, these gaps all disappeared—and high-quality comments declined across the board. In percentage terms, the decline in high-quality comments of 21%–23% (Figure II, Panel B) exceeds the overall decline in comment volume of 18.3% (Figure I, Panel A). Thus, the comments that remain are of lower predicted quality (Online Appendix Table A.5b): 2.9 percentage points fewer comments are helpful ($p = .039$); 1.7 percentage points fewer explain their reasoning ($p = .094$); 1.7 percentage points fewer are actionable ($p = .17$), and 1.9 percentage points fewer likely change the code ($p = .072$).

V.C. Heterogeneous Effects of Proximity on Feedback

Feedback can help build both firm-specific and general human capital. Less-tenured engineers stand to gain more from firm-specific knowledge—such as familiarity with proprietary tools—while younger engineers stand to gain most from general advice, such as how to structure code. We would consequently expect both groups to receive more feedback than their more experienced counterparts. This is exactly what we see on co-located teams. By contrast, when teams are distributed, inexperienced engineers receive little more feedback than their more experienced colleagues.

Figure III, Panel A illustrates the heterogeneity by firm tenure. When offices are open, less-tenured engineers receive more feedback, particularly on co-located teams. When the offices close, feedback declines and converges to a uniformly lower level regardless of tenure. The drop is sharpest for junior engineers on co-located teams, who lose 2.03 more comments per program than junior engineers on already-distributed teams ($p = .001$). When engineers are distributed, less-tenured engineers have less opportunity to learn about the firm.

Figure III, Panel B shows similar patterns by engineer age. When the offices close, younger engineers cease to receive more feedback than their older colleagues. Those on co-located teams are more acutely affected by the closures, losing 2.47 more comments per program than young engineers on already-distributed

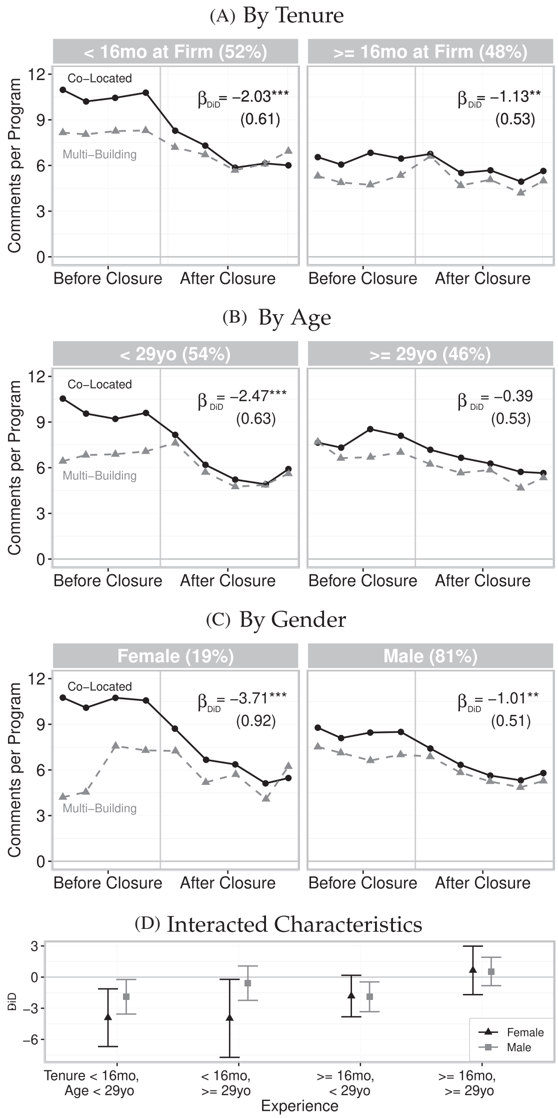

FIGURE III

Heterogeneous Effects of Proximity

This figure replicates Figure I, Panel A by (Panel A) tenure (relative to the mean of 16 months), (Panel B) age (relative to the mean of 29), and (Panel C) gender. Tenure and age are measured at the beginning of each month. Panel D shows the DiD coefficients from a regression with all the interactions. All specifications use our preferred controls and cluster by engineering team. * $p < .1$; ** $p < .05$; *** $p < .01$.

teams (p = .0001). The age heterogeneity persists after controlling for the interaction of tenure and proximity (Online Appendix Table A.7).

1. By Gender. Women receive more feedback on their code than men, but only when they are co-located with their whole team (Figure III, Panel C). When the offices are open, women on co-located teams receive more feedback than their male teammates, while women on multi-building teams receive less feedback than the men on their teams. When the offices close, female engineers on co-located teams lose 3.71 (38.9%) more comments per program than female engineers on already-distributed teams ($p < .0001$). This gender interaction persists after controlling for both tenure and age (Online Appendix Table A.7, columns (5) and (6)).

Men are not inured to the effects of losing proximity, however. They lose 1.01 comments per program (13.1%, $p = .047$). For men, the effects are concentrated among younger engineers (gray squares in Figure III, Panel D, first three columns). For women, the effects extend to older engineers who are new to the firm (black triangle, second column). Among engineers who are neither young nor new to the firm, proximity has no detectable effect on feedback for either men or women (fourth column).

Women appear more reluctant to solicit feedback remotely. When engineers lose proximity to their teammates, women receive feedback from 14.7% (0.26) fewer people per program ($p = .0078$, see Online Appendix Figure A.7) compared with a negligible decline of 2.6% (0.05) for men (gender difference $p = .0056$). When women lose proximity to teammates, they stop asking as many people for feedback, whereas men continue much as before.

One might worry that the additional feedback women receive when sitting near their teammates reflects "mansplaining" rather than mentorship. Several patterns argue against this interpretation. First, feedback from both male and female colleagues declines when proximity is lost (Online Appendix Figure A.8, Panel A). Second, women receive fewer rude comments when they are co-located with teammates than when they are not (Online Appendix Figure A.8, Panel B).[8] Third, like younger and less-tenured engineers, women disproportionately lose high-quality comments—predicted to be helpful, well reasoned, actionable, and

8. We evaluate comments' rudeness using the same process as for their substance (see Online Appendix C).

impactful—when proximity is lost (Online Appendix Figure A.9). Thus, the additional feedback women receive in person appears to be a benefit, not a burden.

In sum, these patterns suggest that losing proximity makes it particularly hard for less-tenured, younger, and female engineers to learn.

VI. Proximity and Code Quality

Proximity increases substantive feedback, which we would expect to improve code quality, especially for engineers who have the most to learn. These improvements could come through two mechanisms. The first is immediate: more detailed reviews from nearby teammates could catch errors and improve the structure of the code. The second operates over time: repeatedly receiving more substantive feedback may help engineers learn and write better code even when they are working apart from their teammates.

VI.A. Immediate Effects

To analyze immediate effects, we turn to the RTO design, since code quality metrics are unavailable around the office closures. Figure IV, Panel A shows that the RTOs increased proximity more for co-located teams than for distributed ones: during the second RTO, for example, co-located engineers worked near 27% of their teammates on a typical day versus 13% for geographically distributed teams.[9] The contrast is starker when considering days with the whole team present: 58% of co-located teams had at least one fully in-person day per month versus just 1% of geographically distributed teams. These fully in-person days may be especially valuable for cohesion, given the negative externalities from even a single distant teammate (Section V.B).

Exploiting this differential increase in proximity, we find that co-location improves code quality for inexperienced engineers. Figure IV, Panel B examines "disposable" code—files that are eventually deleted, either because they were poorly written or misdirected. During the fully remote period, rates of disposable

9. In the second RTO, engineers on co-located teams went to the office on 41% of weekdays. On those days, 64% of their teammates were in, meaning they were with 27% of their teammates on average ($= 100\% \times 0.41 \times 0.64$). When distributed engineers went to the office on 39% of weekdays, only 33% of their teammates were in, so they were with 13% of their teammates on average ($= 100\% \times 0.39 \times 0.33$).

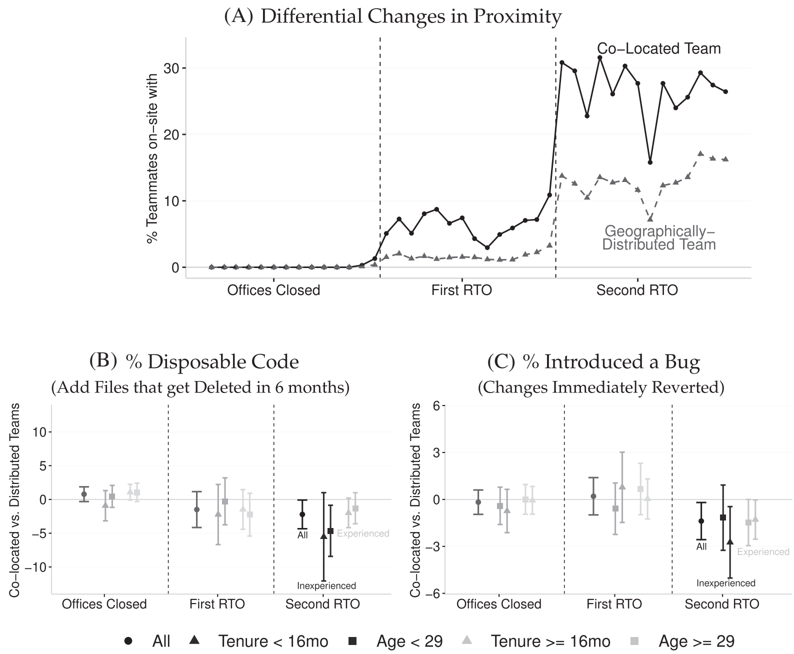

FIGURE IV

Return to Office and Code Quality

This figure illustrates changes around the firm's return-to-office (RTO) phases, comparing engineers in the headquarters (HQ) who are on teams that are all HQ-based versus those that are distributed. Panel A plots face-to-face time with teammates: for each weekday, the measure is zero if the worker is out of the office and otherwise is the share of her teammates also in the office. Panels B and C show code quality differences between engineers on HQ-based and geographically distributed teams with controls for engineer age, tenure, engineering group, and engineer fixed effects. Experience is measured both by tenure at the firm (using a 16-month cutoff) and age (using a 29-year-old cutoff), measured for each engineer at the beginning of the month. Error bars represent 95% confidence intervals, with standard errors clustered by team.

code were similar across co-located and geographically distributed teams, consistent with office assignments being irrelevant when the offices were closed. The same holds during the first RTO, which generated only modest increases in proximity. During the second RTO, however, engineers on co-located teams added 2.2 percentage points fewer files that were later deleted than engineers on geographically distributed teams ($p = .041$). This improvement is concentrated among engineers who have the most to learn, with a 5.5 percentage point improvement for less-tenured engineers (dark gray triangle, $p = .097$) and a 4.6 percentage point improvement for young engineers (dark gray square, $p = .016$). By contrast, older and more tenured engineers show minimal differences across team types (light gray square and triangle, respectively).

Similar patterns emerge for bugs, defined as programs that are immediately and fully reverted after introducing a problem (Figure IV, Panel C). During the second RTO, engineers on co-located teams were 1.4 percentage points less likely to introduce bugs than those on geographically distributed teams ($p = .022$), a gap significantly larger than in either the first RTO or closure period (Online Appendix Table A.10, Panel C, columns (1) and (2)). The effect was nearly twice as large for engineers who were new to the firm at 2.7 percentage points ($p = .019$).

We probe the robustness of these code-quality results to alternative controls. The results on disposable code are nearly identical with the full suite of controls (Online Appendix Figure A.10, Panel B). For the rarer outcome of introducing bugs, adding the full suite of controls does not appreciably change the point estimates but reduces their precision (Online Appendix Figure A.10, Panel C).

VI.B. Long-Run Effects

To assess the long-run effects of proximity on code quality, we examine whether pre-closure exposure to co-located teams predicts code quality once everyone is working remotely and receiving comparable feedback. These specifications all control for engineer age, tenure, and engineering group. Engineers who had been on co-located teams were 2.37 percentage points less likely to write disposable code ($p = .013$, Online Appendix Figure A.4a) and 3.09 percentage points less likely to introduce bugs during the closures ($p = .0012$, Online Appendix Figure A.4b). Much of these gaps persist when we include current team fixed effects: when two engineers work on the same team, the one who had been on a co-located team before the closures tends to write higher-quality code (Online Appendix Table A.6). Consistent with human capital accumulation, code quality improves monotonically with the number of pre-closure months spent on co-located teams (see Online Appendix Figure A.5).

Together, these results suggest that engineers who sit with their teammates receive more feedback, which builds their skills and enables them to write higher-quality code in both the short- and long-run.

VII. Proximity and Code Quantity

Feedback can build engineers' skills, but at what cost? And who bears this burden? Feedback disproportionately comes from experienced engineers, and it is their comments that decline when proximity is lost (Figure V, Panel A). Giving such feedback is time-intensive: it requires reading and understanding the code, diagnosing potential issues, suggesting changes, and explaining the reasoning behind them. Losing proximity might therefore free senior engineers to write more code themselves.

The data support the hypothesis that proximity to teammates has an opportunity cost for senior engineers. Figure V, Panel B shows that while the offices were open, senior engineers on co-located teams wrote 0.76 fewer programs per month for the firm's main codebase, consistent with mentorship consuming a meaningful share of their time ($p = .0005$). Once the offices closed, this difference disappeared—and senior engineers who lost proximity to their teammates saw a relative increase in output of 0.58 programs per month ($p = .0014$). Junior engineers showed no such pattern: they wrote as many programs on co-located teams as distributed ones before the offices closed. Once the offices closed, however, junior engineers previously on co-located teams wrote more programs, suggesting that prior feedback increased their human capital and sustained their long-run productivity.

For all engineers, our DiD design indicates that losing proximity to teammates increases immediate output by 0.48 programs per month ($p = .0002$, Online Appendix Table A.8, column (1)). We see similar effects for other metrics of output, including total lines of code written and total number of files changed, both overall and by seniority (columns (3)–(6)). Results are similar for alternative specifications in Online Appendix Table A.9. The relative changes in productivity are more gradual than for feedback but are still unusual compared to other changes over this period.

We can further analyze whether differences in code quantity reemerged after the offices reopened. Figure VI shows that during the fully remote period and the first RTO, productivity was similar regardless of team type. In the second RTO, this dynamic shifted: as co-located engineers started spending more time with their teammates, they wrote 0.91 (11.7%) fewer programs per

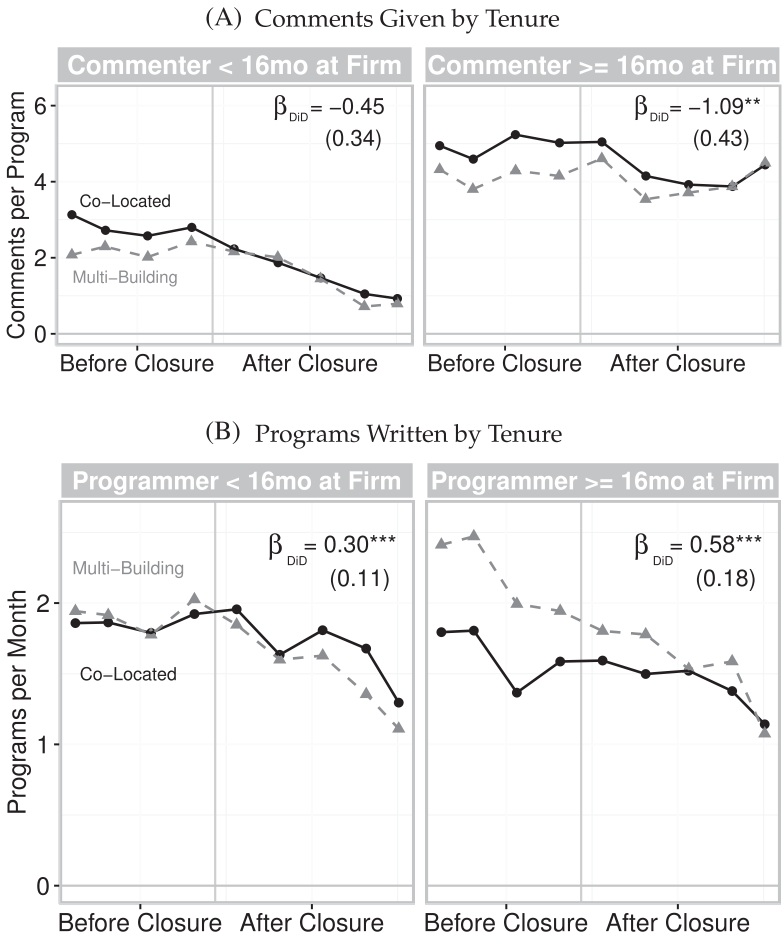

FIGURE V

Trade-offs of Proximity

This figure illustrates the trade-offs of proximity for engineers of different tenures. Panel A shows comments per program broken down by the seniority of the commenter rather than the program writer. Panel B shows programs written per month. Both plots residualize by our preferred controls for engineer age, tenure, and engineering group, as well as program scope (quartics in the number of lines added, lines deleted, and files changed) in Panel A. In each plot, tenure is measured at the beginning of each month. The annotated coefficients report the pooled DiD estimate (equation (1)) for the respective subsample, with standard errors clustered by team in parentheses. * $p < .1$; ** $p < .05$; *** $p < .01$.

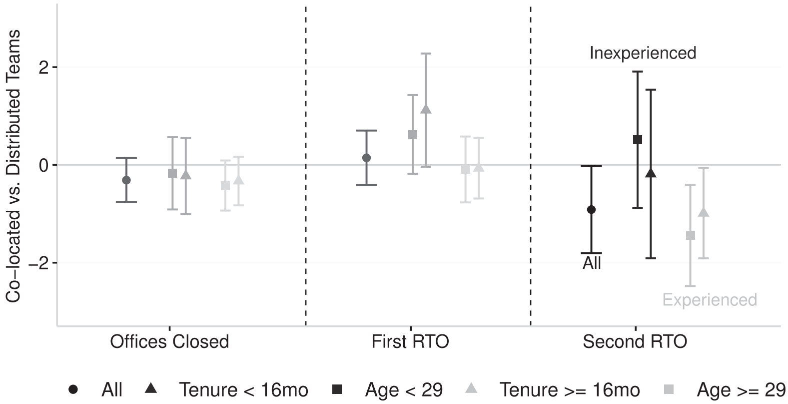

FIGURE VI

Return to Office and Code Quantity

This figure illustrates changes around the firm's return-to-office (RTO) phases, comparing engineers in the headquarters (HQ) who were on teams that are all HQ-based versus those that are distributed. The plot shows differences in programs written per month between engineers on HQ-based and geographically distributed teams with controls for engineer age, tenure, engineering group, and engineer fixed effects. Experience is measured both by tenure at the firm (using a 16-month cutoff) and age (using a 29-year-old cutoff), measured at the beginning of the month for each engineer. Error bars represent 95% confidence intervals, with clustering by team.

month than those on geographically-distributed teams ($p = .044$). This pattern is robust to using the full suite of controls (Online Appendix Figure A.10, Panel A) and analyzing alternative measures of output like lines added (Online Appendix Table A.11).

This decline in programming quantity around the RTOs is driven by experienced engineers, in stark contrast to the code-quality improvements concentrated among juniors. Among engineers with more than 16 months of tenure (the light gray triangles), those on co-located teams wrote fewer programs during the second RTO, while no significant difference emerged for less-tenured engineers (dark gray triangles). Analogously, for engineers who were at least 29 years old (the light gray squares), those on co-located teams wrote 1.4 fewer programs per month during the second RTO ($p = .0064$), Online Appendix Table A.10, Panel A, columns (4) and (5), while no detectable difference emerged for younger engineers (dark gray squares). These patterns suggest that co-location imposes an opportunity cost on experienced engineers so, when not sitting near junior colleagues, they get more done.

Together, these results point to a central trade-off: proximity accelerates junior engineers' human capital development but costs senior engineers time. These findings resonated with engineers at the firm. When we presented our findings internally, one senior engineer described them as "a punch in the gut," explaining that she felt more productive at home but had worried that it was because she was mentoring less. To her—and the other nodding heads in the Zoom room—our results confirmed this concern.

VIII. Who Comes in to the Office?

If young engineers have the most to gain from proximity, we would expect them to come into the office more, and they do. At our partner firm, engineers under the age of 29 were 8.8 percentage points (37.6%) more likely to come into the office during the two RTOs than older engineers when on co-located teams (solid line in Figure VII, Panel A).[10] This difference is roughly halved on geographically distributed teams, where coming in cannot deliver proximity to the whole team (difference $p = .0085$). Young engineers come in both for their teammates and their managers: co-located managers raise attendance by 2.6 percentage points, but the pull of co-located teammates is even stronger at 5.1 percentage points (Online Appendix Table A.13). Similar patterns hold for new hires (Online Appendix Table A.14) and persist after adjusting for commute time and parental status (Online Appendix Table A.15).

These results suggest that those with the most to learn have a particular demand for face-to-face mentorship, although other factors (such as demand for socializing) may also play a role.

Age differences in office attendance extend throughout the tech sector. Figure VII, Panel B draws on survey data from more than 20,000 software engineers collected by Stack Overflow, the leading Q&A site for software developers (Stack Overflow 2023). Nearly half of engineers under age 25 are in the office each day, compared to a quarter of older engineers ($p < .00001$). Within

10. The oldest engineers—who are likely doing the most mentoring—also show elevated attendance on co-located teams, producing a U-shaped pattern by age. For engineers over 40, the gap between co-located and distributed teams is 5.7 percentage points ($p = .0004$).

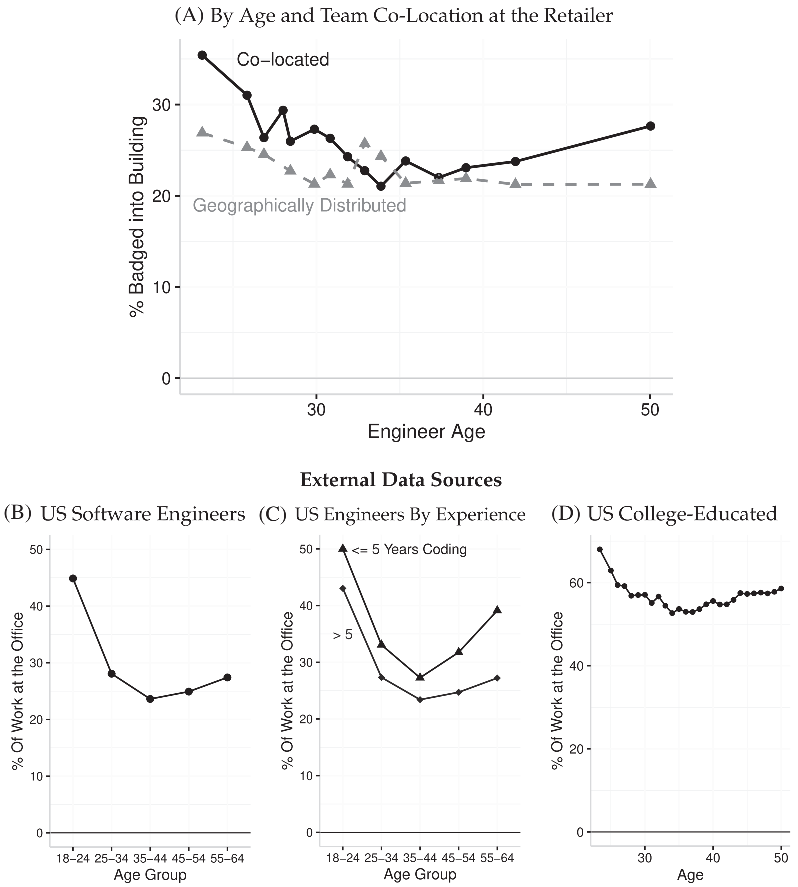

FIGURE VII

Who Returns to the Office

This figure illustrates office attendance patterns by age. Panel A plots office attendance by engineer age at the firm. Data comes from badge swipes in the headquarter's (HQ's) security system, and the sample focuses on engineers based in the HQ during both RTOs from spring of 2022 through spring of 2024. The y-axis represents the percent of weekdays people badge in. The x-axis plots age, split into 15 quantiles. The black series includes engineers whose teammates were all HQ-based; the gray series includes those with some teammates who were not based in the HQ. Panels B and C use data from a 2022 and 2023 survey conducted by Stack Overflow ($N = 22{,}233$) (Stack Overflow 2023). Panel D plots data from the Census's Household Pulse Surveys from 2022 and 2023 (U.S. Census Bureau 2023), limiting to college-educated workers. The y-axis plots the share of reported in-office time.

age groups, engineers with less than five years of coding experience are 5.8 percentage points more likely to be on-site than their same-age peers with more experience (Figure VII, Panel B, $p <$ .00001), consistent with those who have the most to learn having the strongest incentive to show up.

The pattern extends beyond tech. Census Household Pulse data show higher office attendance among young college-educated workers (U.S. Census Bureau 2023) (Figure VII, Panel C). This pattern persists when limiting to nonparents, indicating that it is not simply driven by parents being more likely to work from home (Online Appendix Figure A.15). While many forces may contribute, the results are consistent with a few complementary mechanisms: young workers are more likely to choose to go into the office to build their skills, firms are more adamant that young workers come into the office, and firms are more willing to hire young workers into roles where they can be mentored in person.

IX. Who Is Hired?

Our results indicate that without proximity to coworkers, it is harder to develop talent. A natural response to reduced proximity would be to buy talent built elsewhere, by hiring more experienced engineers. Figure VIII, Panel A shows that the firm does exactly this. When the offices were closed (in black), the firm hired older engineers, effectively buying talent built at other firms. By contrast, when the offices were open—both pre-closure (in light gray) and post-RTO (in dark gray)—the firm hired younger engineers, who may have needed to build their human capital at the firm. Indeed, Figure VIII, Panel B shows that the whole age distribution of hires shifted: over half of hires were under the age of 29 both before closure and after reopening, compared with less than a third during the closure.

Geographic variation in the firm's hiring tells a similar story. For headquarters jobs, most new hires' teammates are local, making proximity feasible when offices are open. For other jobs—which are fully remote or in satellite campuses—teammates are almost always distant.[11] When the offices are open, the firm's headquarters hires are 7–10 years younger than those hired into

11. Indeed, 98% of engineers hired outside the headquarters campus have teammates assigned to other offices.

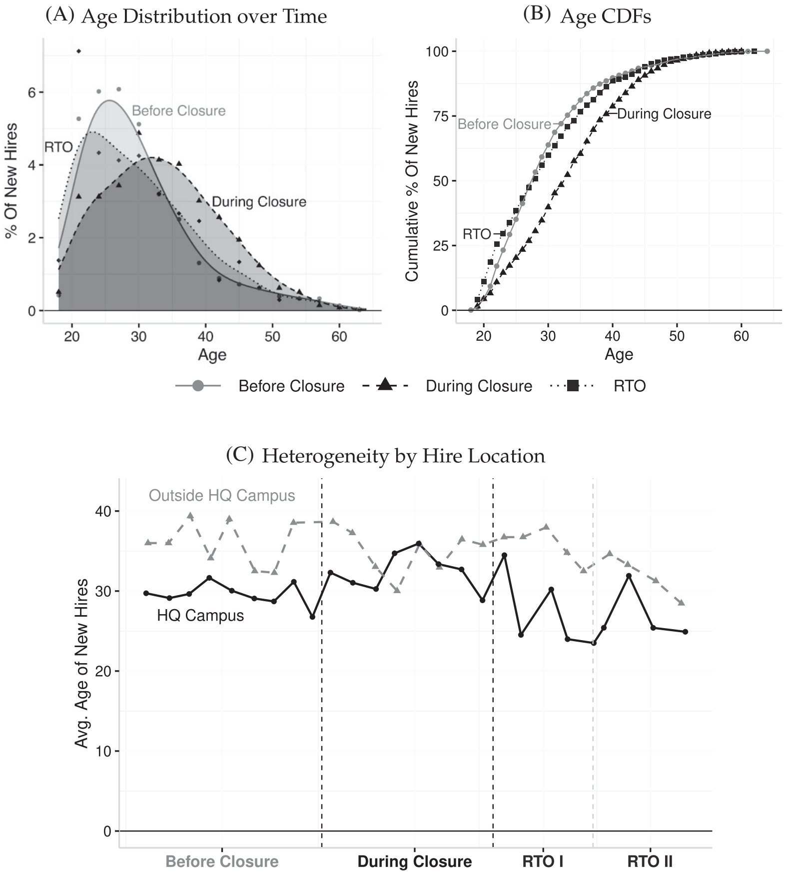

FIGURE VIII

Younger Hires When Proximity Is Feasible

This figure illustrates the shifting age distribution of new hires among the tech workers at the firm. Panel A shows the distribution of new hires; the kernel densities are Gaussian densities with bandwidths of three years and the points show average densities in three-year age-groups. Panel B shows the cumulative distribution functions. Both plots differentiate between the pre-closure period (in light gray), the closure period (in black), and the reopening and return-to-office (RTO) period (in dark gray). Panel C focuses on heterogeneous trends by the location of the new hire. The solid black line shows the pattern for those hired in the headquarters (HQ) campus, who would likely sit with their teammates when the offices were open. The dashed gray line shows the pattern for those hired outside the HQ campus, who would likely be distant from at least some teammates regardless of whether the offices were open (98% of these engineers have teammates assigned to other offices).

distributed roles (Figure VIII, Panel C). By contrast, during the office closures—when everyone was far from their teammates—this age gap narrowed substantially, as the firm hired older workers everywhere.[12] This pattern extends beyond tech to the firm's broader corporate population (Online Appendix Figure A.12).

While supply-side factors could contribute to these patterns, these results are consistent with proximity shifting the relative demand for younger versus older engineers. When the firm cannot ensure physical proximity—because of the timing or location of the job—it hires older workers who have already acquired skills at other firms.

IX.A. Poaching

We can also investigate other firms' hiring decisions by looking at who gets poached from our firm. We define engineers as being poached if they voluntarily leave and say that they are going to a better job.[13] During the office closures, engineers who had accumulated more human capital were more attractive to other firms: 1.2% of co-located engineers were poached each month, compared with 0.9% of multi-building engineers of similar tenure, age, and engineering group (difference $p = .044$, Figure IX, Panel A). These differences compounded over time: by the end of the closure, almost a quarter of co-located engineers had been poached, versus a sixth of multi-building engineers. We see no corresponding differences in firings, layoffs, or other types of quits (Online Appendix Figure A.13). Consistent with the differences reflecting accumulated human capital, we estimate a significant dose response: engineers who spent more of their pre-closure time on co-located teams were more likely to be poached after the offices closed (Online Appendix Table A.12).

The poaching differences are concentrated among engineers who are building more general human capital. Younger engineers on co-located teams receive feedback that likely builds transferable skills and are disproportionately poached (Figure IX, Panel B). Female engineers on co-located teams similarly receive substantially more feedback and also disproportionately land better

12. The patterns are qualitatively similar when excluding remote workers and only focusing on those hired in the main versus satellite campuses (Online Appendix Figure A.11).

13. Using data from Glassdoor, we see that poached engineers do typically go to positions that pay more.

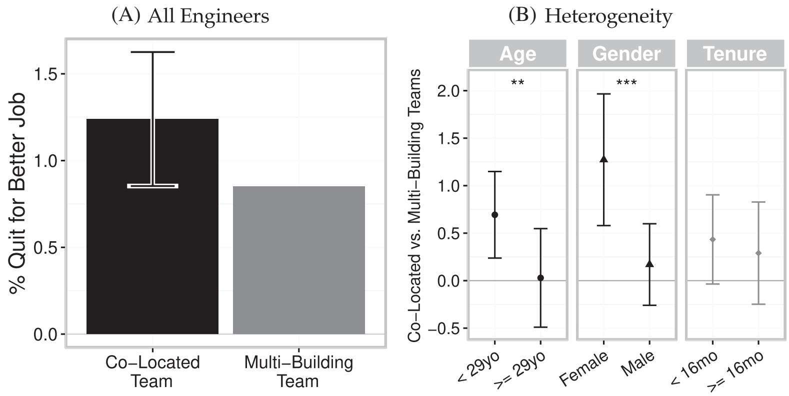

FIGURE IX

Poaching from the Firm During the Office Closures

This figure investigates whether engineers who were on co-located teams and received more human capital investments were more likely to be poached by other firms. The analysis focuses on when offices were closed and proximity was impossible, so other firms may have been more inclined to hire those with more accumulated human capital. Panel A shows the differences between engineers on co-located versus multi-building teams. Engineers on co-located teams were consistently co-located in the pre-closure period. Panel B investigates the differences between co-located and multi-building teams for different subgroups of engineers—of different age, gender, and tenure. The annotation indicates the significance of the difference across the two groups (e.g., between younger and older engineers). All specifications include our preferred time-varying controls for engineer group, age, and tenure, as well as gender (where applicable). Standard errors are clustered by team. * $p < .1$; ** $p < .05$; *** $p < .01$.

jobs elsewhere. Tenure, by contrast, does not moderate the effect, consistent with less-tenured engineers' feedback being more firm-specific.

One reason the firm may struggle to retain these engineers is that its compensation system rewards relative performance within teams rather than absolute skill. As a result, the firm may fail to price in the human capital advantage that co-location confers. Consistent with this, total compensation is similar for engineers on co-located and multi-building teams both before and after the office closures (Online Appendix Figure A.14).

Poaching erodes the firm's returns on human capital investments. Thus, even if the firm could perfectly observe coworkers' investments in each other's human capital, it would still under-incentivize them (Becker 1964). Social bonds between proximate coworkers may help fill this void and mitigate underinvestment in general human capital.

X. Who Can't Find a Job?

This section uses large-scale survey data to show suggestive evidence that our firm's hiring practices are not unique: the rise of remote work has made it relatively harder for young people to find jobs. We compare unemployment trends in remotable jobs (like software engineering or marketing) to non-remotable jobs (like mechanical engineering or nursing) using Dingel and Neiman (2020)'s occupational classification.[14] We focus on college-educated workers who are more similar to the software engineers whom we study (Online Appendix D shows results for noncollege graduates).

Young college graduates in remotable occupations have found it more difficult to find jobs since the pandemic, whereas older workers in the same occupations have fared better. Between 2017–2019 and 2022–2024, young graduates (under age 29) in remotable occupations experienced a 0.88 percentage point increase in unemployment ($p < .00001$), while older graduates saw a marginal decline of 0.11 percentage points ($p = .053$).[15] The resulting widening of the age gap in unemployment coincided with the pandemic and has remained elevated alongside rates of remote work (Figure X, Panel A).[16]

In non-remotable occupations, by contrast, young graduates' relative unemployment rate spiked up briefly in 2020 before returning to baseline.

Figure X, Panel B presents results from a triple-difference regression comparing the change in the age gap in unemployment

14. Hansen et al. (2023) show that some remotable occupations in Dingel and Neiman (2020)'s classification are rarely remote in practice, such as teachers and managers in in-person sectors. Our results are robust to dropping these potentially misclassified occupations (Online Appendix Figure A.16).

15. We focus on the age threshold of 29 years because that is what we have used throughout the article. However, results are similar with alternative definitions. Online Appendix Figure A.17, Panel A shows that unemployment rates systematically increased for young workers in remotable jobs, while declining for older workers.

16. While our partner firm shifted back to hiring younger workers after its RTOs in 2022–2023, many employers of remotable workers have continued to offer more remote-work flexibility. Additionally, some RTOs do not always meaningfully increase proximity because teams are geographically distributed (Goldberg 2021). Indeed, for such distributed positions, our partner firm hires older workers.

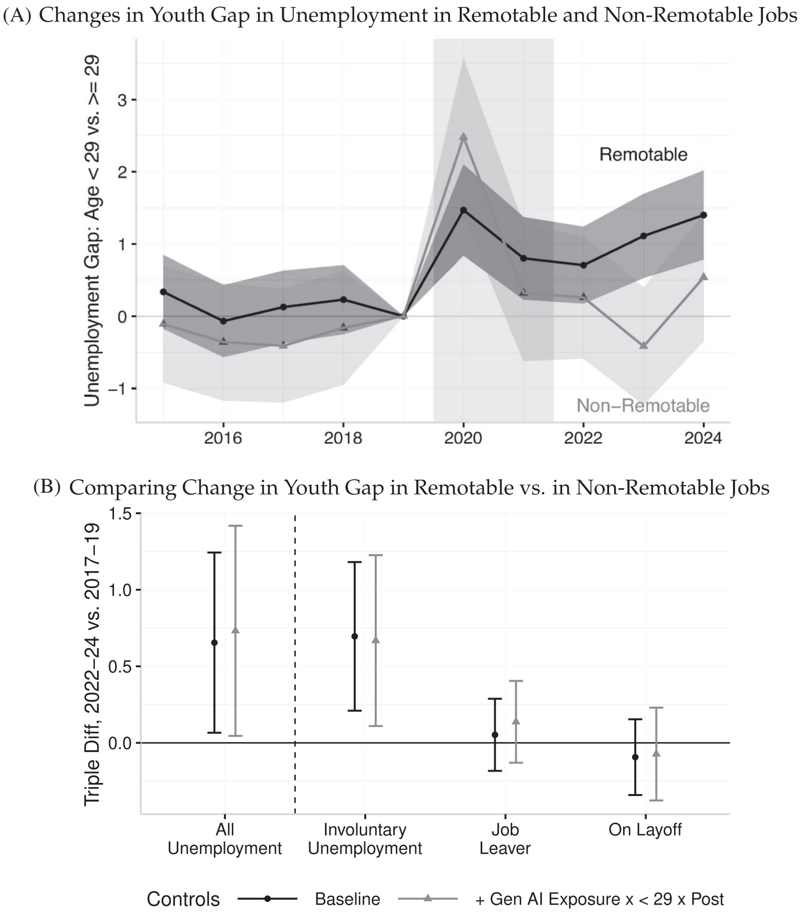

FIGURE X

The Rise of Remote Work and Young People's Unemployment

This figure analyzes changes in young people's relative unemployment rates around the rise of remote work. Panel A shows dynamic changes in the youth gap in unemployment (between those < 29 and ≥ 29) relative to 2019, with occupation and age fixed effects. These changes are shown separately for those in remotable jobs (in black circles) and non-remotable jobs (in gray triangles). Online Appendix Figure A.17, Panel B shows a continuous version of this analysis. Panel B shows triple-difference coefficients comparing the change in the youth gap in unemployment in remotable versus non-remotable jobs, excluding the peak pandemic years of 2020–2021 (equation (4)). Baseline controls include age, occupation, and year fixed effects; additional controls fully interact occupational exposure to generative AI (Eisfeldt, Schubert, and Zhang 2023) with indicators for being < 29 and the post period (2022–2024). Data are from the Current Population Survey (Flood et al. 2024), limited to college graduates between ages 22 and 64. Standard errors are clustered by survey respondent.

across remotable and non-remotable occupations.[17] Young college graduates in remotable occupations experienced a 0.65 percentage point greater increase in unemployment than would have been expected based on trends among either older workers in remotable occupations or young workers in non-remotable ones ($p = .029$). This differential is robust to controlling for occupations' exposure to generative AI and its potentially heterogeneous effects across age groups, using the occupational index in Eisfeldt, Schubert, and Zhang (2023), which aggregates up from Eloundou et al. (2024)'s task-level exposures (the coefficients in triangles in Figure X, Panel B). This figure also shows that the differential increase is driven by involuntary and persistent forms of unemployment, not voluntary job leaving nor temporary layoffs.

A back-of-the-envelope calculation indicates that between 2017–2019 and 2022–2024, remote work can explain 64% of the increase in unemployment among all young college graduates, many of whom enter remotable jobs. For this calculation, we scale our triple-difference estimate (0.65 percentage points) by the share of young graduates in remotable jobs (61%): this would predict a 0.4 percentage point increase in all young college graduates' unemployment, which is 64% of the realized increase of 0.63 percentage points.

While many have attributed the recent labor market challenges of young college graduates to generative AI (e.g., Roose 2025), the uptick in their unemployment rate predates the rapid diffusion of AI and is concentrated more among those in remotable occupations than those whose tasks are most amenable to automation. Of course, generative AI may play a more defining role as it diffuses further, and other factors may be contributing to young graduates' labor market struggles. As a result, the unemployment patterns should be interpreted with caution. Still, the evidence

17. The triple-difference coefficient is β in:

$$\begin{aligned}\text{Unemployed}_{it} = {} & \beta \mathbb{1}[\text{Age}_{it} < 29] \times \text{Remotable}_i \times \text{Post}_t \\ & + \delta \mathbb{1}[\text{Age}_{it} < 29] \times \text{Post}_t + \sigma \text{Remotable}_i \times \text{Post}_t \\ & + \psi \mathbb{1}[\text{Age}_{it} < 29] \times \text{Remotable}_i + \mu_a + \mu_o + \mu_t + v_{it}, \end{aligned} \tag{4}$$

where μ_a denotes age fixed effects; μ_o, occupation fixed effects; and μ_t, year fixed effects. We exclude the height of the pandemic in 2020–2021 and instead compare 2022–2024 to 2017–2019.

suggests that the rise of remote work meaningfully contributes to the recent challenges of young graduates.

XI. Conclusion

This article shows that physical proximity between coworkers meaningfully increases mentorship and skill development—even among software engineers, the quintessential digital natives with an array of virtual communication tools. Face-to-face interaction is particularly valuable for younger and less experienced workers, who receive more feedback from senior colleagues when co-located and ultimately write higher-quality code. Thus, an important side effect of remote work's rise may be its scarring effects on young workers, who struggle to learn from colleagues in an increasingly distributed world.

But mentorship is not free: it impedes the output of experienced workers who do the mentoring. Firms thus face a trade-off: remote work enhances productivity today, as experienced workers focus on their own output, while sacrificing productivity tomorrow, as junior workers receive less mentorship and develop fewer skills.

Remote work stands to shift who is hired, with adverse effects for young workers. Our retailer hired fewer young workers when employees were distant from their teammates during the office closures and when hiring outside the headquarters campus. This mirrors hiring patterns nationally, where young workers' unemployment rates rose alongside the rise of remote work, particularly in remotable occupations.

Our results indicate that remote work may have nuanced effects on gender equity. Although it may enable working mothers to remain in the workforce (Harrington and Kahn 2025; Ho, Jalota, and Karandikar 2026; Jalota and Ho 2024), it may be costly for young women, whose professional development appears to be especially sensitive to physical proximity to colleagues.

One worker cannot achieve the gains from proximity by going into the office alone: their teammates must do the same. Moreover, even one distant teammate can degrade connections among those who are physically together, and short distances can have outsized effects on coworker interactions. Realizing the full returns to in-person work thus requires not just individual presence but collective, coordinated attendance.

Supplementary Material

An Online Appendix for this article can be found at *The Quarterly Journal of Economics* online.

Data Availability

The publicly available data and code underlying this article are available in Emanuel, Harrington, and Pallais (2026), in the Harvard Dataverse, https://doi.org/10.7910/DVN/OHFHXZ.

Federal Reserve Bank of New York, United States
University of Virginia, United States
Harvard University and the National Bureau of Economic Research, United States

References

Agrawal, Ajay, and Avi Goldfarb, “Restructuring Research: Communication Costs and the Democratization of University Innovation,” *American Economic Review*, 98 (2008), 1578–1590. https://doi.org/10.1257/aer.98.4.1578.

Akan, Mert, Jose Maria Barrero, Nicholas Bloom, Tom Bowen, Shelby Buckman, Steven J Davis, Luke Pardue, and Liz Wilke, “Americans Now Live Farther from Their Employers,” Daily report, Hoover Institution, Stanford, CA, 2024.

Allen, Thomas, and Gunter Henn, *The Organization and Architecture of Innovation* (London: Routledge, 2007). https://doi.org/10.4324/9780080545370.

Angelici, Marta, and Paola Profeta, “Smart Working: Work Flexibility Without Constraints,” *Management Science*, 70 (2024), 1680–1705. https://doi.org/10.1287/mnsc.2023.4767.

Atkin, David, M. Keith Chen, and Anton Popov, “The Returns to Face-to-Face Interactions: Knowledge Spillovers in Silicon Valley,” Working Paper no. 30147, National Bureau of Economic Research, Cambridge, MA, 2022. https://doi.org/10.3386/w30147.

Battiston, Diego, Jordi Blanes i Vidal, and Tom Kirchmaier, “Face-to-Face Communication in Organizations,” *Review of Economic Studies*, 88 (2021), 574–609. https://doi.org/10.1093/restud/rdaa060.

Becker, Gary S., *Human Capital: A Theoretical and Empirical Analysis, with Special References to Education* (Chicago: University of Chicago Press, 1964).

Belden, Martha Elaine, “Has WFH Changed the Way We See?,” All About Vision, 2022. https://www.allaboutvision.com/conditions/computer-vision-syndrome/have-screens-changed-vision/.

Bird, Steven, Ewan Klein, and Edward Loper, *Natural Language Processing With Python: Analyzing Text with the Natural Language Toolkit* (Sebastopol, CA: O’Reilly Media, 2009).

Bloom, Nicholas, Ruobing Han, and James Liang, “Hybrid Working from Home Improves Retention Without Damaging Performance,” *Nature*, 630 (2024), 920–925. https://doi.org/10.1038/s41586-024-07500-2.

Bloom, Nicholas, James Liang, John Roberts, and Zhichun Jenny Ying, “Does Working from Home Work? Evidence from a Chinese Experiment,” *Quarterly Journal of Economics*, 130 (2015), 165–218. https://doi.org/10.1093/qje/qju032.

Brynjolfsson, Erik, Bharat Chandar, and Ruyu Chen, “Canaries in the Coal Mine?: Six Facts about the Recent Employment Effects of Artificial

Intelligence," Working Paper, Stanford Digital Economy Lab, Palo Alto, CA, 2025. https://digitaleconomy.stanford.edu/wp-content/uploads/2025/11/CanariesintheCoalMine_Nov25.pdf.

Catalini, Christian, "Microgeography and the Direction of Inventive Activity," *Management Science*, 64 (2018), 4348–4364. https://doi.org/10.1287/mnsc.2017.2798.

Chen, Tianqi, and Carlos Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York: ACM, 2016), 785–794.

Choudhury, Prithwiraj, Cirrus Foroughi, and Barbara Larson, "Work-from-Anywhere: The Productivity Effects of Geographic Flexibility," *Strategic Management Journal*, 42 (2021), 655–683. https://doi.org/10.1002/smj.3251.

Choudhury, Prithwiraj, Christos A Makridis, Tarun Khanna, and Kyle Schirmann, "Is Hybrid Work the Best of Both Worlds? Evidence from a Field Experiment," *Review of Economics and Statistics* , forthcoming. https://doi.org/10.1162/rest_a_01428.

Cullen, Zoe B., Bobak Pakzad-Hurson, and Ricardo Perez-Truglia, "Home Sweet Home: How Much Do Employees Value Remote Work?," Working Paper no. 33383, National Bureau of Economic Research, Cambridge, MA, 2025. https://doi.org/10.3386/w33383.

de Groot, Jasper H. B., Monique A. M. Smeets, Matt J. Rowson, Patricia J. Bulsing, Cor G. Blonk, Joy E. Wilkinson, and Gün R. Semin, "A Sniff of Happiness," *Psychological Science*, 26 (2015), 684–700. https://doi.org/10.1177/0956797614566318.

Dingel, Jonathan I., and Brent Neiman, "How Many Jobs Can Be Done at Home?" *Journal of Public Economics*, 189 (2020), 104235. https://doi.org/10.1016/j.jpubeco.2020.104235.

Dutcher, E. Glenn, "The Effects of Telecommuting on Productivity: An Experimental Examination. The Role of Dull and Creative Tasks," *Journal of Economic Behavior & Organization*, 84 (2012), 355–363. https://doi.org/10.1016/j.jebo.2012.04.009.

Eisfeldt, Andrea L., Gregor Schubert, and Miao Ben Zhang, "Generative AI and Firm Values," Working Paper no. 31222, National Bureau of Economic Research, Cambridge, MA, 2023. https://doi.org/10.3386/w31222.

Eloundou, Tyna, Sam Manning, Pamela Mishkin, and Daniel Rock, "GPTs Are GPTs: Labor Market Impact Potential of LLMs," *Science*, 384 (2024), 1306–1308. https://doi.org/10.1126/science.adj0998.

Emanuel, Natalia, and Emma Harrington, "Working Remotely? Selection, Treatment, and the Market for Remote Work," *American Economic Journal: Applied Economics*, 16 (2024), 528–559. https://doi.org/10.1257/app.20230376.

Emanuel, Natalia, Emma Harrington, and Amanda Pallais, "Replication Data for: 'The Power of Proximity to Coworkers'," 2026, Harvard Dataverse. https://doi.org/10.7910/DVN/OHFHXZ.

Fenizia, Alessandra, and Thomas Kirchmaier, "Not Incentivized Yet Efficient: Working From Home in the Public Sector," Discussion Paper no. 2036, Centre for Economic Performance, London School of Economics and Political Science, London, 2024.

Flood, Sarah, Miriam King, Renae Rodgers, Steven Ruggles, Robert J. Warren, Daniel Backman, Annie Chen, Grace Cooper, Stephanie Richards, Megan Schouweiler, and Michael Westberry, "IPUMS CPS: Version 12.0" [datset], 2024. https://cps.ipums.org/cps/.

Gaspar, Jess, and Edward L. Glaeser, "Information Technology and the Future of Cities," *Journal of Urban Economics*, 43 (1998), 136–156. https://doi.org/https://doi.org/10.1006/juec.1996.2031.

Gibbs, Michael, Friederike Mengel, and Christoph Siemroth, "Work from Home and Productivity: Evidence from Personnel and Analytics Data on Information Technology Professionals," *Journal of Political Economy Microeconomics*, 1 (2023), 7–41. https://doi.org/10.1086/721803.

Goldberg, Emma, "The Worst of Both Worlds: Zooming from the Office," *The New York Times*, Nov. 16, 2021. https://www.nytimes.com/2021/11/16/business/return-to-office-hybrid-work.html.
Goodman-Bacon, Andrew, "Difference-in-Differences with Variation in Treatment Timing," *Journal of Econometrics*, 225 (2021), 254–277. https://doi.org/10.1016/j.jeconom.2021.03.014.
Hansen, Stephen, Peter John Lambert, Nicholas Bloom, Steven J. Davis, Raffaella Sadun, and Bledi Taska, "Remote Work Across Jobs, Companies, and Space," Working Paper no. 31007, National Bureau of Economic Research, Cambridge, MA, 2023. https://doi.org/10.3386/w31007.
Harrington, Emma, and Matthew E. Kahn, "Has the Rise of Work-From-Home Reduced the Motherhood Penalty in the Labor Market," Working Paper no. 34147, National Bureau of Economic Research, Cambridge, MA, 2025. https://doi.org/10.3386/w34147.
He, Haoran, David Neumark, and Qian Weng, "Do Workers Value Flexible Jobs? A Field Experiment," *Journal of Labor Economics*, 39 (2021), 709–738. https://doi.org/10.1086/711226.
Herkenhoff, Kyle, Jeremy Lise, Guido Menzio, and Gordon M. Phillips, "Production and Learning in Teams," *Econometrica*, 92 (2024), 467–504. https://doi.org/10.3982/ECTA16748.
Ho, Lisa, Suhani Jalota, and Anahita Karandikar, "Bringing Work Home: Flexible Arrangements as Gateway Jobs for Women in West Bengal," Working Paper, Structural Transformation and Economic Growth, Growth Research Platform, London, 2026.
Hosseini Maasoum, Seyed Mahdi, and Guy Lichtinger, "Generative AI as Seniority-Biased Technological Change: Evidence from US Résumé and Job Posting Data," available at SSRN, 2025. http://dx.doi.org/10.2139/ssrn.5425555.
Jalota, Suhani, and Lisa Ho, "What Works for Her? How Work-from-Home Jobs Affect Female Labor Force Participation in Urban India," available at SSRN, 2024. https://doi.org/10.2139/ssrn.4739387.
Kraut, Robert, Carmen Egido, and Jolene Galegher, "Patterns of Contact and Communication in Scientific Research Collaboration," in *Proceedings of the 1988 ACM Conference on Computer-Supported Cooperative Work* (New York: ACM, 1988), 1–12.
Lewandowski, Piotr, Katarzyna Lipowska, and Mateusz Smoter, "Mismatch in Preferences for Working from Home: Evidence from Discrete Choice Experiments With Workers and Employers," ILR Review, 79 (2026), 142–172. https://doi.org/10.1177/00197939251355928.
Maestas, Nicole, Kathleen J. Mullen, David Powell, Till von Wachter, and Jeffrey B. Wenger, "The Value of Working Conditions in the United States and Implications for the Structure of Wages," *American Economic Review*, 113 (2023), 2007–2047. https://doi.org/10.1257/aer.20190846.
Mas, Alexandre, and Amanda Pallais, "Valuing Alternative Work Arrangements," *American Economic Review*, 107 (2017), 3722–3759. https://doi.org/10.1257/aer.20161500.
———, "Alternative Work Arrangements," *Annual Review of Economics*, 12 (2020), 631–658. https://doi.org/10.1146/annurev-economics-022020-032512.
Roose, Kevin, "For Some Recent Graduates, the A.I. Job Apocalypse May Already Be Here," *The New York Times*, May 30, 2025. https://www.nytimes.com/2025/05/30/technology/ai-jobs-college-graduates.html.
Ruggles, Steven, Sarah Flood, Matthew Sobek, Daniel Backman, Annie Chen, Grace Cooper, Stephanie Richards, Renae Rogers, and Megan Schouweiler, "IPUMS USA: Version 14.0" [dataset], 2022. https://usa.ipums.org/usa/.
Salazar Miranda, Arianna, and Matthew Claudel, "Spatial Proximity Matters: A Study on Collaboration," *PLoS One*, 16 (2021), e0259965. https://doi.org/10.1371/journal.pone.0259965.

Schubert, Gregor, “Organizational Technology Ladders: Remote Work and Generative AI Adoption,” available at SSRN, 2025. https://dx.doi.org/10.2139/ssrn.5094265.

Schwartz, Linoy, Jonathan Levy, Yaara Endevelt-Shapira, Amir Djalovski, Olga Hayut, Guillaume Dumas, and Ruth Feldman, “Technologically-Assisted Communication Attenuates Inter-Brain Synchrony,” *NeuroImage*, 264 (2022), 119677. https://doi.org/10.1016/j.neuroimage.2022.119677.

Seltzer, Leslie J., Ashley R. Prososki, Toni E. Ziegler, and Seth D. Pollak, “Instant Messages vs. Speech: Hormones and Why We Still Need to Hear Each Other,” *Evolution and Human Behavior*, 33 (2012), 42–45. https://doi.org/10.1016/j.evolhumbehav.2011.05.004.

Stack Overflow, “Stack Overflow Developer Survey 2017–2023,” 2023. https://survey.stackoverflow.co/.

U.S. Census Bureau, “Household Pulse Survey,” 2023. https://www.census.gov/data/experimental-data-products/household-pulse-survey.html.

Wang, Baolian, Peixin Li, and Jiawei Yu, “The Impact of Return-to-Office Mandates on Equity Analysts,” available at SSRN, 2025. https://dx.doi.org/10.2139/ssrn.5206058.

Yang, Longqi, David Holtz, Sonia Jaffe, Siddharth Suri, Shilpi Sinha, Jeffrey Weston, Connor Joyce, Neha Shah, Kevin Sherman, and Brent Hecht et al., “The Effects of Remote Work on Collaboration Among Information Workers,” *Nature Human Behaviour*, 6 (2022), 43–54. https://doi.org/10.1038/s41562-021-01196-4.