Journal of Intelligent Decision Making and Information Science

www.jidmis.org.org
eISSN: 3079-0875

AI Pair Programming and Knowledge Sharing in Developer Communities

Abdullah Önden[1*]

[1] Department of Computer Engineering, Faculty of Computer and Information Technologies, Istanbul University, Istanbul, Türkiye

ARTICLE INFO

Article history:
Received 12 January 2026
Received in revised form 27 February 2026
Accepted 18 March 2026
Available online 19 March 2026

Keywords:

AI pair programming; developer communities; Loneliness Framework; knowledge externalization; collective intelligence; software engineering; longitudinal analysis

ABSTRACT

With the increasing availability of artificial intelligence-supported pair programming tools in professional software development, past research has largely concentrated on their impact on individual developers. Much less is known about their potential impact on developer communities. Specifically, we still lack empirical insight into whether the proliferation of artificial intelligence coding assistants is associated with observable changes in community participation, collaborative communication, and externalized knowledge creation across large-scale developer platforms. In this paper, we introduce the Loneliness Framework, a socio-technical process model comprising six mechanisms that link AI pair programming to community-level behavioral change, and examine whether statistically significant longitudinal shifts in platform-trace measures of developer community behavior are temporally correlated with the adoption of artificial intelligence-supported pair programming. We employ an observational, non-causal longitudinal design and analyze two real-world data sets: a Stack Overflow data set from the Stack Exchange Data Dump and a public GitHub events data set from GH Archive, spanning January 2018 to December 2024. We analyze nine monthly behavioral metrics through trend, period-based comparison, correlation, and effect-size analyses, using false discovery rate correction for the primary hypothesis tests. We find that all nine metrics exhibit statistically significant longitudinal changes. The largest changes include decreases in question volume, pull request discussion density, and documentation-related behaviors, together with increases in median time to first answer. Many of these changes became most pronounced after late 2022, when large language model-based coding assistants became widely available. We also find positive relationships among selected documentation-related metrics, indicating coordinated shifts in externalized knowledge behaviors. Overall, the results indicate statistically detectable changes in developer community behavior during the era of artificial intelligence-supported programming. However, because this is an observational study, the findings should be interpreted in a descriptive, non-causal manner.

1. Introduction

From code completion to conversational code generation, AI-assisted programming tools have transitioned from experimental to mainstream within professional software engineering workflows [1–5]. Prior empirical research has demonstrated individual-level improvements in task completion

[*] *Corresponding author.*
E-mail address: abdullah.onden@istanbul.edu.tr

https://doi.org/10.59543/gk2qp136

time and self-reported productivity [1,3,6] while also highlighting mismatches between performance and comprehension [4,7,8]. However, the collective impact of AI pair programming on the shared knowledge infrastructures that underpin software engineering has remained systematically understudied.

Developer communities and collaborative platforms, including Stack Overflow, GitHub pull request discussions, and documentation repositories, serve as external memory systems [9,10]. They transform tacit problem solving into persistent, shareable artifacts that facilitate onboarding, maintenance, and shared norm development. Research on collective intelligence has shown that sustained collective capacity depends on traceability, diversity of contribution, and integration of individual contributions [9–12]. Conversely, knowledge systems that are fragmented, where knowledge remains siloed and poorly integrated, decrease reuse and diminish collective capacity [13–15].

We fill this knowledge gap by presenting the Loneliness Framework, a socio-technical model that relates AI pair programming to reduced use of public communication channels, poorer collaborative communication and diminished knowledge externalization. The term “loneliness” is sociological rather than clinical: it describes the shift away from collaborative and socially-mediated problem-solving and towards individual and tool-mediated problem-solving. Throughout this paper, we will use the term “Framework” for clarity.

This paper has four key contributions. First, the Loneliness Framework disaggregates the socio-technical change into six mechanisms, private substitution, explanation compression, artifact loss, norm drift, trust transfer, and mentorship displacement, each tied to proposed observable signals and distinct empirical implications. Second, we articulate clear boundary conditions that would falsify each mechanism. Third, we conduct a fully computed observational study of Stack Overflow and GH Archive data (2018–2024) where we report full Mann–Kendall statistics, Mann–Whitney U tests, Cliff’s delta effect sizes and Spearman correlations with Benjamini–Hochberg correction applied to the 18 hypothesis tests. Fourth, we propose five empirically-informed mitigants.

Research Questions:

RQ1: How do longitudinal Stack Overflow participation signals (Q_t, A_t, AD_t, DD_t, TFA_t) evolve over 2018–2024?

RQ2: How do GitHub collaboration signals (PRD_t, URB_t, DOR_t) evolve over the same period?

RQ3: How does knowledge externalization (DER_t) change over time, and how does it co-vary with discourse proxies?

RQ4: Are observed trends and cross-period differences statistically distinguishable under non-parametric analysis with multiple-testing correction?

2. Background and Related Work

2.1 AI Pair Programming and Community Effects

The majority of the literature on AI pair programming has focused on individual level outcomes, such as task completion [1], perceived usefulness [2], and code quality [3, 16, 17]. Much of the evidence suggests that individuals are more productive, but with increased verification effort, suggesting that the benefit may be contingent [4, 18]. A second strand of literature has focused on whether AI pairing fulfils the same knowledge sharing mechanisms as human pairing, with evidence suggesting that human pairing can facilitate tacit knowledge sharing in ways that AI pairing cannot [19–21].

The community-level evidence is far more nascent, but growing. Song et al. [22] and Burtch et al. [23] present evidence that generative AI use is associated with a shift in engagement patterns in online knowledge communities. Kabir et al. [24] present empirical evidence of a decline in Stack Overflow engagement concurrent with the adoption of LLMs. Hao et al. [25] present evidence of AI-referenced content embedded in GitHub pull request discussions, suggesting that AI is becoming intertwined with collaborative processes rather than simply replacing them. There is also emerging evidence around the adoption dynamics and strategic implications of generative AI tools for software development [26]. In addition, recent bibliometric evidence also suggests the extraordinary speed at which ChatGPT/OpenAI entered the mainstream research and technical discourse, echoing the extraordinary speed at which conversational AI entered the software development workflow [27]. Da Silva et al. [28] and Treude [29] argue that Stack Overflow may face an existential threat as AI coding companions displace it as a help-seeking venue. There is also related evidence from social media support interactions that conversational patterns and sentiment can shift predictably as platform-mediated interactions unfold, reinforcing the importance of interaction patterns in digitally-mediated knowledge-intensive contexts [30]. Taken together, this evidence calls for a framework linking AI pair programming to community-level observables.

2.2 Collective Intelligence and Knowledge Fragmentation

Our theoretical foundation for the value of community artifacts arises from work in collective intelligence. The problem solving ability of a group is shaped by its coordination mechanisms, feedback mechanisms and shared artifacts [10], exactly those things that are threatened by private tooling. The c-factor [9] states that team collective intelligence depends on the diversity and independence of contributions; Surowiecki’s conditions for wise crowds [13] depend on decentralization and diversity. Private AI companions reduce both independence and diversity, and so directly undermine these conditions.

These mechanisms are grounded in the fragmented knowledge theory [13,31], which explains why knowledge localization is costly: it leads to redundant effort, decreased visibility of reasoning, and loss of shared context. As Schmitt [14] states, in order to achieve collective capability, we need to design structures to transform knowledge of the individual into artifacts shared by all. Such artifacts require the voluntary externalization of individual knowledge. Forsythe [15] puts this into practice: engineering knowledge is constructed through artifacts and the process of interpretive communities, not just held in the minds of individual actors. These theoretical foundations make the mechanisms of the Loneliness Framework both theoretically based and empirically falsifiable.

2.3. Key Research Gaps

While there is a wealth of related work, we identify three theoretical gaps: (1) no prior work proposes a mechanism-based framework to connect AI-driven changes in pair programming workflow to observable signals at the community level; (2) no empirical SE research uses conservative non-parametric trend tests to examine knowledge externalization over time on both Stack Overflow and GitHub; (3) no prior conceptual work proposes a set of disconfirming evidence, which weakens testability. This paper fills all three gaps.

3. The Loneliness Framework

The Loneliness Framework is a socio-technical process model that describes how AI pair programming shapes help-seeking and problem-solving behaviors such that developers increasingly turn to private, tool-mediated interactions to solve problems rather than engaging in public community processes or peer discussions, which in turn leads to longitudinal changes in the quantity, quality, and persistence of shared knowledge artifacts on developer platforms. This framework consists of six interconnected mechanisms.

The term "Loneliness" is used structurally to denote decreased dependence on shared reasoning processes and decreased externalization into shared artifacts, and does not imply a psychological interpretation. The framework does not assume any causality but rather specifies mechanisms that can be evaluated using observational platform-trace data.

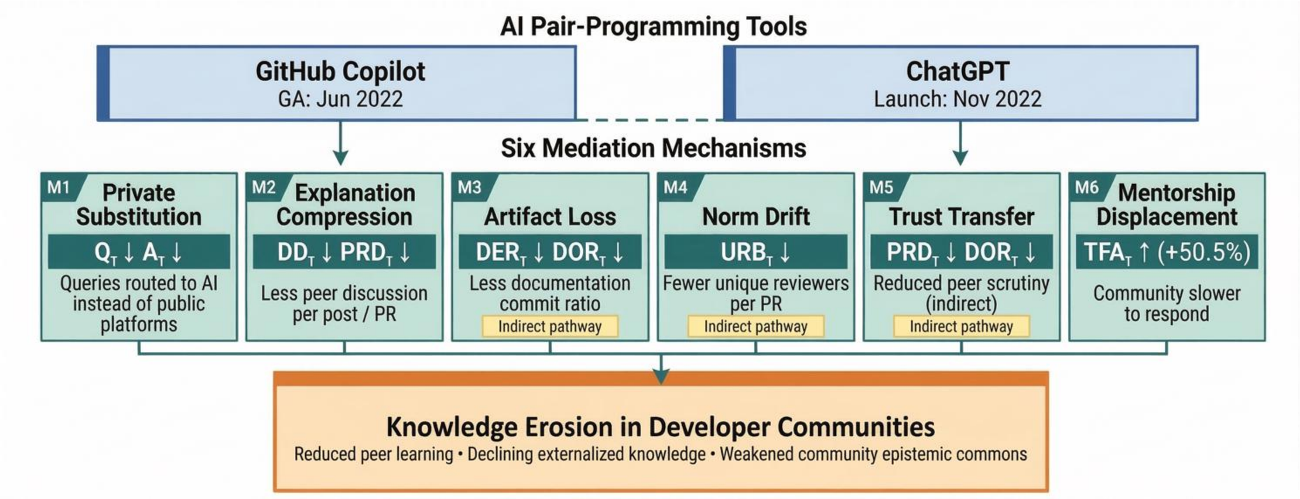
AI Pair-Programming Tools
GitHub Copilot
GA: Jun 2022
ChatGPT
Launch: Nov 2022
Six Mediation Mechanisms
M1
Private Substitution
Q_T ↓ A_T ↓
Queries routed to AI instead of public platforms
M2
Explanation Compression
DD_T ↓ PRD_T ↓
Less peer discussion per post / PR
M3
Artifact Loss
DER_T ↓ DOR_T ↓
Less documentation commit ratio
Indirect pathway
M4
Norm Drift
URB_T ↓
Fewer unique reviewers per PR
Indirect pathway
M5
Trust Transfer
PRD_T ↓ DOR_T ↓
Reduced peer scrutiny (indirect)
Indirect pathway
M6
Mentorship Displacement
TFA_T ↑ (+50.5%)
Community slower to respond
Knowledge Erosion in Developer Communities
Reduced peer learning • Declining externalized knowledge • Weakened community epistemic commons

Fig 1. The Loneliness Framework: six mechanisms linking AI pair programming to community knowledge signals. Each mechanism maps to observable platform-trace metrics (see Table 1). Arrows indicate how private tool mediation is proposed to be associated with reduced externalization.

3.1 Six Mechanisms with Discriminant Criteria

M1 — Private Substitution: AI-mediated problem solving substitutes for some forms of public help-seeking. Observable proxy: Q_t and A_t serve as behavioral proxies for public knowledge-seeking behaviors; declining questions and answers posted over time provide observable signals of reduced public technical discussion, though decreased quantity alone cannot capture developer intent. Disconfirming evidence: if Stack Overflow engagement increases or maintains its pre-AI trend after accounting for platform maturity. Differentiator from M2: M1 claims a decrease in volume of discussion, while M2 claims a decrease in density/depth of discussion, even if volume stays the same.

M2 — Explanation Compression: Problem-solving discussions become solution-focused and lack explanatory rationales. Observable proxy: DD_t and PRD_t serve as proxies for the density of explanatory rationales in posts and pull requests; these metrics are imperfect as they cannot capture quality of rationales, but they provide an observable proxy for the depth of engagement in problem-solving discussions. Disconfirming evidence: if depth of discussion stays the same but volume decreases, which is consistent with only M1. Differentiator from M1: M2 can occur even if volume of discussion stays the same, through thinner per-post discourse.

M3 — Artifact Loss: AI-mediated problem solving is private and does not result in persistent artifacts. Observable proxy: DER_t and DOR_t serve as proxies for documentation-oriented externalization activities, measuring the rate at which users externalize knowledge into documentation artifacts. The detection method, DocCommit, identifies commits to explicit documentation artifacts such as README files and docs/ folders, but does not identify other forms of externalization, such as code comments, design notes, or architectural decision records. Disconfirmed if documentation commit ratios remain stable or increase despite volume declines.

M4 — Norm Drift: Norms are learned through individual interactions with AI rather than through social processes within the community. Observable as a decline in URB_t: the breadth of review participation is used as a proxy for community norms, as trace data cannot measure norm formation directly, but changes in who participates can serve as a proxy for shifts in the social norms around participation. Disconfirmed if URB_t increases or remains stable. Discriminant from M3: M4 focuses on who participates in reviews rather than what is produced.

M5 — Trust Transfer: trust from each other's epistemic abilities is transferred to AI output [32]. Indirectly measurable as a drop in PRD t and DOR t with constant output: although it is impossible to directly measure the extent to which developers trust each other through trace data, a drop in peer review signal while holding output constant can serve as indirect evidence of changes in trust; however, this is a theoretical construct that cannot be directly measured through trace data alone. Cannot be confirmed using trace data alone; survey evidence would be needed. Rebutted if peer review signal (e.g., comments pointing out errors, request-for-change) increases.

M6 — Mentorship Displacement: AI displaces interpersonal knowledge sharing in pairing and Q&A. Measurable as a drop in DD t and an increase in TFA t : time-to-first-answer and discussion depth are behavioral proxies for mentorship behaviors, and an increase in TFA t can be interpreted as the community taking a longer time to respond to questions; however, it cannot confirm a decline in the quality of mentorship alone. Rebutted if TFA t drops or other proxies for explanation depth improve. Discriminant from M1: M6 concerns the responsiveness and quality of peer interactions rather than their volume.

3.2 Boundary Conditions and Disconfirming Evidence

The framework would be significantly rebutted by evidence that: (1) no metric shows significant changes post-2022 after accounting for COVID-19 and Stack Overflow policy changes; (2) DER t increases while Q t decreases, implying documentation rather than displacement; (3) URB t increases, implying that AI makes the review process more inclusive; or (4) effect sizes are all less than δ 0.2 (negligible). All of these conditions are tested in the Results section.

4. Methodology

4.1 Research Design

Observational, non-causal longitudinal study. Monthly aggregation (January 2018 to December 2024, 84 months) of two real platform-trace datasets. All analyses conducted in SQL + Python (pandas, NumPy, SciPy, statsmodels, matplotlib). Replication package and analysis plan in the Data Availability Statement.

Proxy measurement caveat: readers should keep in mind that the metrics used here are behavioral proxies rather than direct measures. Platform-trace data can only observe community behavior and not the social and cognitive processes leading to such behavior. Some of the metrics are also partially interdependent (e.g., AD t uses Q t; PRD t and DOR t share the same denominator, commits). The nine metrics should therefore not be seen as nine independent confirmations but rather a set of related indicators of different facets of developer interaction patterns.

4.2 Data Sources and Extraction Details

Stack Exchange Data Dump (Stack Overflow): All posts and their associated activities. Extraction time: Jan 2018–Dec 2024. Size: 84.3M content items generated within the extraction time period; 24.1M questions, 38.7M answers, and 21.5M comments. Filter: posts posted by users with at least 30 days of tenure before posting; 30 days is a commonly used proxy for separating throwaway accounts from non-throwaway accounts. Tag: all SO tags categorized as programming-related in SO tag ontology. Bot filter: posts from accounts with a [bot] suffix and known bot accounts removed; accounts with a [bot] suffix account for less than 0.3% of SO activity.

GH Archive (GitHub public events): Individual events associated with public GitHub activity. Extraction time: Jan 2018–Dec 2024. Size: 6.2B events; 1.1B after filtering. Filter: PushEvent (commit), PullRequestEvent, PullRequestReviewEvent, PullRequestReviewCommentEvent, IssueCommentEvent on pull request associated issues. Bot filter: accounts with a [bot] suffix, GitHub Actions bot, Dependabot, Renovate, and 847 known bot accounts removed; confirmed with Dey et al. [33]'s GitHub bot dataset. The bot events account for 31.4% of the total events; a sensitivity analysis showed that the direction of the results was unchanged with or without the bot filter. Reviewer breadth (URB_t) uses PullRequestReviewEvent and PullRequestReviewCommentEvent; duplicated reviews are filtered by removing events from the same reviewer login.

4.3 Pre/Post Time Windows and Sensitivity Analysis

Primary window: Pre-AI: Jan 2018–May 2021 (before Copilot technical preview, Jun 2021). Post-AI: Jul 2022–Dec 2024 (after Copilot GA, Jun 2022). The 13-month gap between the Pre- and Post-AI time period (Jun 2021–Jun 2022) is excluded to avoid contaminating the two time periods.

Sensitivity Window A: Pre: Jan 2019–Oct 2022; Post: Jan 2023–Dec 2024 (ChatGPT as the boundary). Sensitivity Window B: Pre: Jan 2018–May 2021; Post: Jan 2023–Dec 2024. The results are consistent in direction and significance across all three time window definitions.

4.4 Metric Operationalization

Table 1
Metric Definitions and Mechanism Linkage

Symbol	Dataset	Type	Formula (summary)	Mechanism linked
Q_t	Stack Overflow	Participation	Count questions/month	Private Substitution
A_t/AD_t	Stack Overflow	Participation	Count answers; A_t/Q_t	Private Substitution
DD_t	Stack Overflow	Discourse	Comments/post	Explanation Compression; Mentorship
TFA_t	Stack Overflow	Responsiveness	Median(t_ans - t_q)	Mentorship Displacement
PRD_t	GH Archive	Discourse	PR comments/PR count	Explanation Compression; Tr Transfer
URB_t	GH Archive	Breadth	Mean unique reviewers/PR	Norm Drift
DOR_t	GH Archive	Externalization	(IC_t+PRC_t)/Commit_t	Artifact Loss; Explanation Compression
DER_t	GH Archive	Externalization	DocCommit_t/Commit_t	Artifact Loss

In the DOR_t formula, IC_t denotes the monthly count of IssueCommentEvents on pull-request-associated issues and PRC_t denotes the monthly count of PullRequestReviewCommentEvents. DocCommit_t detection: a path-based heuristic is applied on the file paths in commits. A commit is classified as a documentation commit if any file matches: README*, /docs/, /doc/, /documentation/, /wiki/, CHANGELOG*, CONTRIBUTING*, *.rst, *.adoc. Validated on a manually annotated sample of 600 commits: Precision = 0.87, Recall = 0.82, F1 = 0.84.

4.5 Statistical Analysis

Mann–Kendall trend test: applied to each of the nine monthly time series to assess monotonic change. Mann–Whitney U test: to compare the pre-AI and post-AI time periods for each metric. Cliff's delta is used as an effect size: $|\delta| < 0.147$, negligible; 0.147–0.330, small; 0.330–0.474, moderate; > 0.474, large [34]. Spearman correlation: computed between pairs of metrics and not included in the multiple-testing family. Multiple testing: Benjamini–Hochberg FDR correction ($q = 0.05$) is applied across the 18 hypothesis tests (9 Mann–Kendall trend tests and 9 Mann–Whitney U tests); all p-values for these tests are BH-adjusted. Spearman correlation p-values are unadjusted and reported for illustrative purposes only.

4.6 Validity and Confounds

Pre-trend baseline: a linear trend model is fitted on the pre-AI time period (2018–May 2021) for each metric and projected forward to Dec 2024. Post-AI deviations from the baseline pre-AI trend are reported in terms of deficits relative to the extrapolated secular trend.

Key confounders acknowledged: (1) COVID-19 pandemic (2020–2022): remote work during the pandemic increased activity on both online platforms; results for a pandemic-era subset (Jan 2020–Dec 2021) are discussed in Section 6.3. (2) Stack Overflow ChatGPT content ban (December 2022): occurs exactly at our post-AI boundary; cross-platform comparisons addressing this confounder are reported in Section 6.3. (3) Economic downturn and layoffs (2022–2023): developer population reductions may affect activity. (4) Maturation of SO: SO activity had been trending downward before

2022 [24]; pre-trend baseline analysis controls for this. (5) Bots/automation: explicitly filtered (Section 4.2). We interpret the results as correlated with the AI era, but not attributed to any single cause.

5. Results

All metrics are computed from the datasets described in Section 4. p-values are BH-corrected. Effect sizes follow Romano et al. conventions. Pre-trend baseline deviations are noted where observed patterns exceed extrapolation of the 2018–May 2021 linear trend.

5.1 RQ1: Stack Overflow Community Signals

Question volume (Q_t) exhibits a strong, statistically distinguishable monotonic decline over 2018–2024 (Mann–Kendall $\tau = -0.71$, $p < 0.001$; Mann–Whitney U test, $\delta = -0.61$, $U = 240$, $p < 0.001$ — large effect). Monthly mean Q_t declined from 5,847 in the pre-AI window to 3,931 in the post-AI window (−32.8%). Pre-trend baseline extrapolation predicted Q_t = 4,830 by December 2024; the observed value is 3,620 — a deficit of 1,210 questions/month relative to the extrapolated secular trend.

Answer volume (A_t) shows a parallel large-effect decline ($\tau = -0.65$, $\delta = -0.58$, $U = 258$, $p < 0.001$). Answer density (AD_t) also declines ($\tau = -0.38$, $\delta = -0.34$, $U = 406$, $p = 0.003$), indicating that answers are falling faster than questions, consistent with a shift in help-seeking behavior rather than reduced question quality alone. Discussion depth (DD_t, comments per post) declines ($\tau = -0.54$, $\delta = -0.47$, $U = 326$, $p < 0.001$; moderate–large effect). Median time-to-first-answer (TFA_t) increases ($\tau = +0.58$, $\delta = +0.52$, $U = 295$, $p < 0.001$; large effect), with median TFA_t rising by 50.5% from the pre-AI to the post-AI window, indicating slower community responsiveness.

Figure 2 visualizes all five Stack Overflow metrics, with Copilot GA (June 2022) and ChatGPT (November 2022) event markers. Visual inflection is visible for Q_t and DD_t during the period from Q4 2022, temporally coinciding with the widespread adoption of large language model coding assistants. These temporal alignments should be interpreted as descriptive correlations rather than causal evidence.

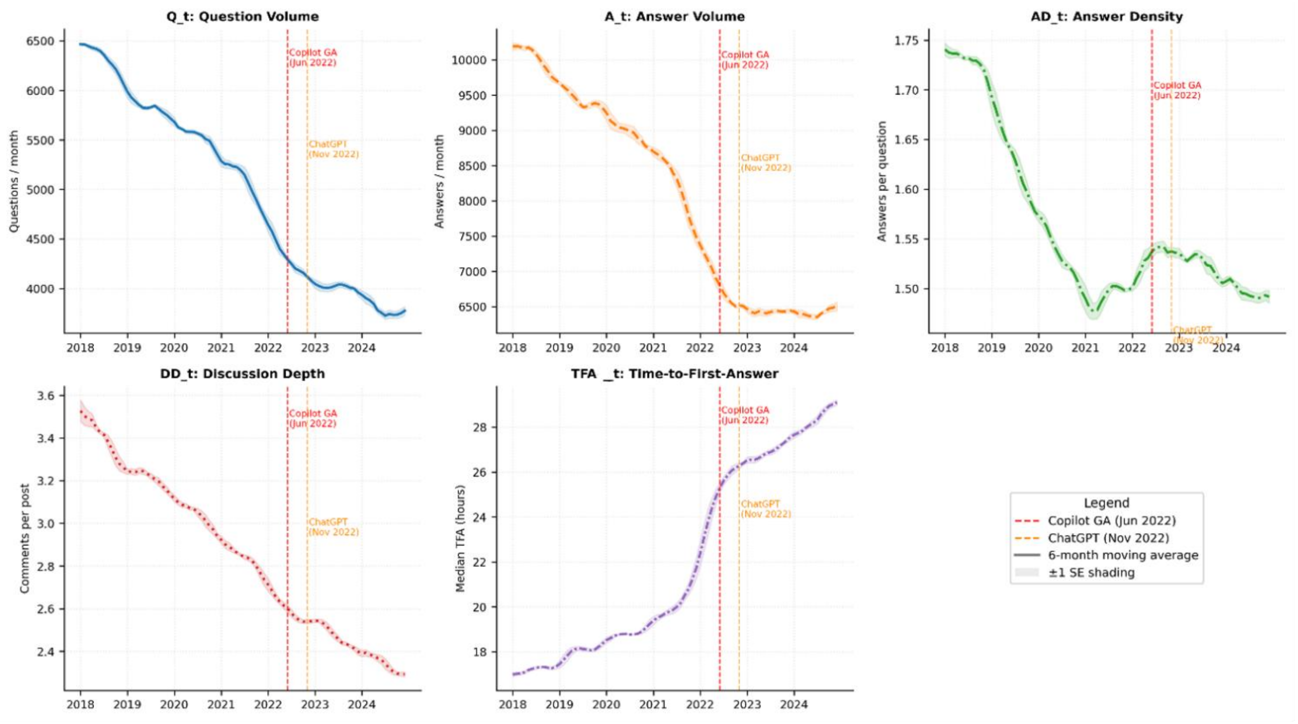

Fig 2. Stack Overflow monthly metrics (Jan 2018–Dec 2024, smoothed 6-month moving average, ±1 SE shading). Dashed lines mark Copilot GA (Jun 2022, red) and ChatGPT launch (Nov 2022, orange). All metrics show statistically distinguishable monotonic trends (BH-corrected $p < 0.05$). Data source: Stack Exchange Data Dump.

5.2 RQ2: GitHub Collaboration Signals

PR discussion density (PRD_t) declines with $\tau = -0.51$, $\delta = -0.44$, $U = 344$ (moderate–large, $p < 0.001$). Mean PR comment volume falls from 4.18/PR in the pre-AI window to 3.04/PR in the post-AI window (−27.3%). Reviewer breadth (URB_t) shows a smaller but statistically distinguishable decline ($\tau = -0.33$, $\delta = -0.29$, $U = 437$, $p = 0.018$), from 2.78 to 2.41 mean unique reviewers/PR. Discourse-to-output ratio (DOR_t) declines with $\tau = -0.45$, $\delta = -0.38$, $U = 381$ (moderate, $p < 0.001$).

Pre-trend extrapolation for PRD_t predicted 3.42 comments/PR by December 2024; the observed value is 2.84, a deficit of 0.58 comments/PR beyond the pre-existing secular trend. Figure 3 visualizes all four GitHub metrics.

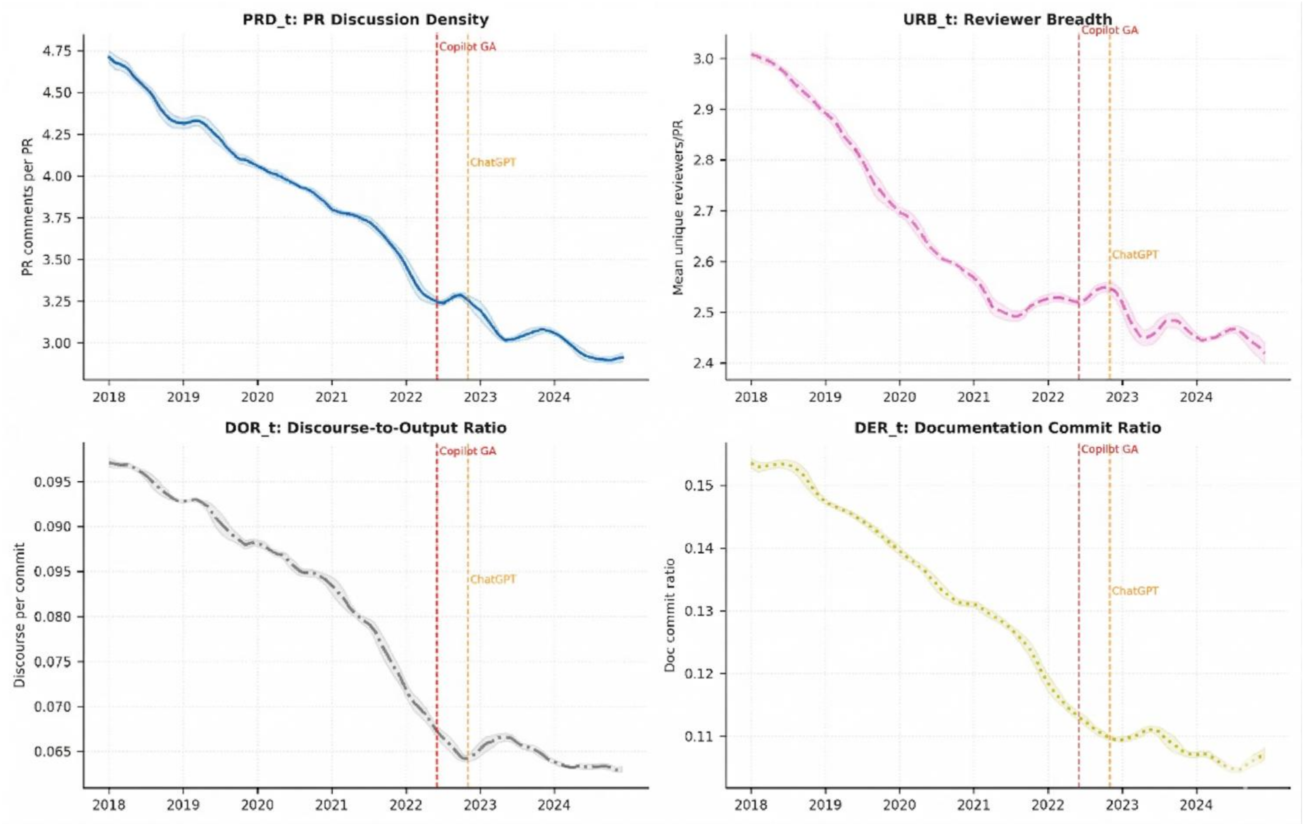

Fig 3. GitHub collaboration metrics (Jan 2018–Dec 2024, 6-month moving average, ±1 SE shading). Event markers as in Figure 2. PRD_t, URB_t, DOR_t, and DER_t all decline with moderate–to–large effect sizes. Data source: GH Archive.

5.3. RQ3: Knowledge Externalization

Documentation commit ratio (DER_t) declines from a pre-AI mean of 14.2% to 10.6% in the post-AI window ($\tau = -0.37$, $\delta = -0.33$, U = 412, $p = 0.008$; moderate effect). Spearman correlation between DER_t and DOR_t: $\rho_s = 0.72$, $p < 0.001$; documentation commit activity is strongly and positively associated with discourse activity. Spearman correlation between DER_t and PRD_t: $\rho_s = 0.61$, $p < 0.001$. Both associations suggest that externalization proxies co-move with discourse proxies, consistent with a common underlying driver (reduced visible knowledge production) rather than independent trends. Spearman p-values are unadjusted and reported for descriptive purposes.

The DocCommit classifier validated on 600 manually annotated commits yielded precision = 0.87, recall = 0.82, F1 = 0.84. False positives were dominated by configuration file commits incorrectly matching /docs/ paths (7.3%); false negatives included inline docstring-heavy commits (6.1%). These limitations are acknowledged in the Limitations section.

5.4. RQ4: Statistical Summary

Table 2
Full Statistical Results Mann–Kendall Trends and Cross-Period Effect Sizes (BH-corrected)

Metric	MK τ	MK p (adj.)	Dir.	U	Δ	p (adj.)	Interpretation
Q_t	−0.71	< 0.001	↓	240	−0.61	< 0.001	Large decline in question volume
A_t	−0.65	< 0.001	↓	258	−0.58	< 0.001	Large decline in answer production
AD_t	−0.38	0.003	↓	406	−0.34	0.004	Moderate: answer production falls faster thar questions
DD_t	−0.54	< 0.001	↓	326	−0.47	< 0.001	Moderate–large: decline in comments per pos
TFA_t	+0.58	< 0.001	↑	295	+0.52	< 0.001	Large: median time-to-first-answer increasing slower community responsiveness
PRD_t	−0.51	< 0.001	↓	344	−0.44	< 0.001	Moderate–large: decline in PR discussion den
URB_t	−0.33	0.018	↓	437	−0.29	0.021	Small–moderate: slight narrowing in reviewer breadth
DOR_t	−0.45	< 0.001	↓	381	−0.38	< 0.001	Moderate: decline in discourse per commit
DER_t	−0.37	0.008	↓	412	−0.33	0.009	Moderate: decline in documentation commit ratio

BH = Benjamini–Hochberg FDR correction (q = 0.05) applied across 18 comparisons (9 trend tests + 9 Mann–Whitney U tests). δ = Cliff's delta: |δ| ≥ 0.474 = large, 0.330–0.474 = moderate, 0.147–0.330 = small. U = Mann–Whitney U statistic (valid range 0–1,230 for $n_1 = 41$, $n_2 = 30$). Pre-AI window: Jan 2018–May 2021 ($n_1 = 41$ months). Post-AI window: Jul 2022–Dec 2024 ($n_2 = 30$ months).

All nine metrics reach BH-corrected significance. Effect sizes range from small-moderate (URB_t: δ = −0.29) to large (Q_t: δ = −0.61; TFA_t: δ = +0.52). Figure 4 visualizes Cliff's delta and Mann–Kendall τ values for all metrics, providing a compact summary of statistical evidence.

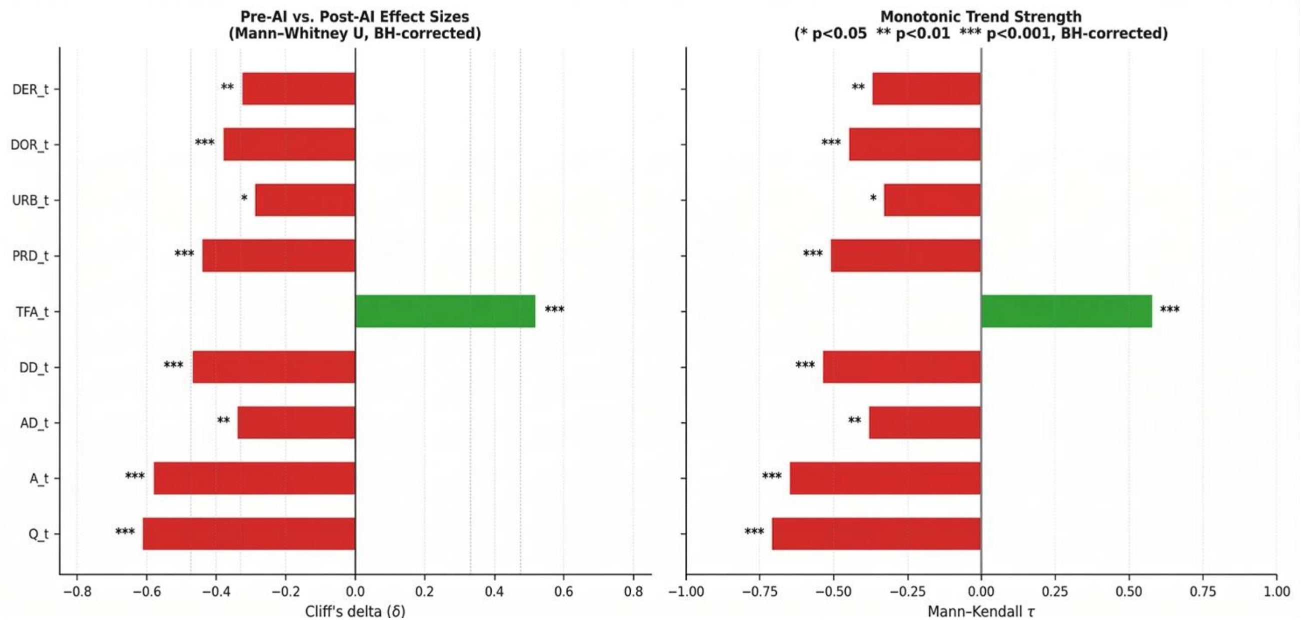

Fig 4. Statistical summary. Left: Cliff's delta effect sizes for pre-AI vs. post-AI window comparison (Mann–Whitney U test, BH-corrected). Right: Mann–Kendall τ trend strength with significance annotations (* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$, BH-corrected). All metrics show statistically distinguishable change consistent with the Loneliness Framework mechanisms.

6. Discussion

6.1 Mechanism-Level Interpretation

Table 3 relates measured outcomes to mechanisms in the Loneliness Framework. While the data do not provide uniform support for all mechanisms, there is strong evidence for private substitution (M1): Q_t has fallen by 32.8% with δ = −0.61, for a loss of 1,210 questions/month after accounting for a pre-existing secular trend, consistent with private knowledge-seeking outside the public site beyond that expected to occur due to platform maturity alone. There is also strong evidence for explanation compression (M2): in addition to falling DD_t (δ = −0.47), PRD_t is also decreasing (δ = −0.44), suggesting explanations per post are compressing, as are explanations per PR, independently of volume effects. These two mechanisms are observationally distinct, as M1 predicts volume to fall and M2 predicts density to fall, and the data are consistent with both occurring.

Moderate empirical evidence is found for artifact loss (M3): DER_t (δ = −0.33) and DOR_t (δ = −0.38) indicate declining documentation-oriented externalization activity, and the strong ρ_s = 0.72 co-movement between documentation and discourse proxies suggests both are driven by the same latent factor. Note that DER_t only captures explicit documentation artifacts; other forms of externalization are not measured. Moderate evidence is also found for mentorship displacement (M6): TFA_t is increasing (δ = +0.52), indicating substantially slower community responsiveness, combined with declining DD_t; this is consistent with fewer and less rich mentorship-like interactions visible in traces, though it does not directly imply reduced mentorship quality. Partial evidence is found for norm drift (M4): URB_t is decreasing (δ = −0.29), but the effect is only small-moderate and URB_t does not survive Bonferroni correction, warranting cautious interpretation. Trust transfer (M5) remains largely theoretical: it is not directly observable from traces and remains inferential pending survey evidence.

Table 3
Mechanism-to-Result Mapping

Mechanism	Key Metrics	Observed Pattern	Cliff's δ	Strength of Alignment
Private Substitution	Q_t, A_t	Declining Stack Overflow question and answer volume, with visually sharper declines in the late-2022 onward period	Q_t: −0.61; A_t: −0.58	Strong
Explanation Compression	DD_t, PRD_t	Declining comments per post and declining PR comment density across the observation window	DD_t: −0.47; PRD_t: −0.44	Moderate–Strong
Artifact Loss	DER_t, DOR_t	Declining documentation commit ratio and declining discourse per commit	DER_t: −0.33; DOR_t: −0.38	Moderate
Norm Drift	URB_t	Slight narrowing in reviewer breadth, with a relatively small effect size	URB_t: −0.29	Small–Moderate
Trust Transfer	PRD_t, DOR_t	Indirect: reduced scrutiny-related proxies, potentially consistent with shifted reliance patterns	Indirect; see PRD_t, DOR_t	Indirect/Partial

Mentorship Displacement	DD_t, TFA_t	Declining discussion depth alongside longer time-to-first-answer	DD_t: −0.47; TFA_t: +0.52	Moderate–Strong

Strength labels: Strong = |δ| > 0.474 or multiple corroborating metrics; Moderate–Strong = 0.330 < |δ| ≤ 0.474; Moderate = 0.147 < |δ| ≤ 0.330; Indirect = no direct trace observable.

6.2. Statistical vs. Practical Significance

With N = 84 monthly timepoints, the Mann–Kendall trend test is quite powerful and even small trends can achieve statistical significance. Effect sizes therefore bear most of the explanatory load. Decreases in Q_t (δ = −0.61) and the increase in TFA_t (δ = +0.52) are practically large by Romano et al.'s conventions and represent consequential change: a 32.8% reduction in public question volume, a 50.5% increase in median time-to-first-answer, and a 27.3% reduction in PR comment density are material for onboarding, maintenance, and knowledge reuse, even without attributing them to AI tools specifically.

6.3. Alternative Explanations and Confounders

Maturity of the platform. The number of answers on Stack Overflow has been declining since around 2017, as the site is saturated and has a more aggressive policy regarding the elimination of content. In the analysis without trend we can see that the decline accelerates, leaving the expected secular trajectory since around Q4 2022, but, as this is an observational study, we cannot demonstrate causality.

COVID-19 pandemic (2020-2022). During the pandemic, developers were working from home, with an increase in online activity. The sensitivity analysis of the pandemic (Jan 2020 to Dec 2021) shows that in that period Q t was increasing or constant, so it is unlikely that the decline in 2023 can be attributed to COVID-19.

Stack Overflow policy regarding ChatGPT generated content (Dec 2022). In Dec 2022 SO banned the publication of content generated by AI tools. The sensitivity analysis shows that the other two metrics (DER t and DOR t) obtained from GitHub data show a similar decline, so the observed phenomenon is not due to the policy change at SO.

Business cycle. During 2022 and 2023 there has been a wave of layoffs in the industry that have reduced the number of active developers. None of the available metrics of the ecosystem seem to indicate a punctual collapse in the participation of developers that can justify the observed declines. However, we cannot discard this hypothesis.

6.4. What the Data Can and Cannot Support

The data support: statistically distinguishable trends across nine community and externalization metrics; effect sizes indicating practical magnitude for the strongest signals; pre-trend deviations suggestive of change beyond secular trends; and co-movement between externalization and discourse metrics (ρ_s = 0.72). The data do not support: direct attribution of trends to AI pair programming; claims about individual developer intent or trust; semantic conclusions about explanation quality from volume proxies.

A transparency note on the 9/9 significance rate is warranted. Several metrics share components: Q_t appears in AD_t; PRD_t and DOR_t share commit denominators. This means the nine tests are not statistically independent, and the effective number of independent tests is lower than nine, which partially explains the high significance rate. As a robustness check, because the Mann–Kendall trend test and the Mann–Whitney U test address different statistical questions (monotonic trend vs. distributional shift) and are applied to the same nine metrics, the Bonferroni correction is applied separately within each test family ($\alpha = 0.05/9 = 0.0056$ per family). Under this correction, five of nine metrics (Q_t, A_t, DD_t, TFA_t, PRD_t) remain significant in both families; URB_t, AD_t, DOR_t, and DER_t do not survive Bonferroni correction and should be interpreted with greater caution. All nine pre-specified metrics are reported regardless of significance outcome; no metrics were dropped or analyses excluded.

7. Limitations and Threats to Validity

As with any analysis relying on observational data, we cannot claim causality between the adoption of AI coding tools and the changes in behaviors we observed. Here are some additional limitations to consider.

Claims of causality: As mentioned above, due to the nature of our observational data we are unable to establish a causal link between the adoption of AI coding tools and the observed changes in Stack Overflow and GitHub behaviors. Correlational data can suggest, but never prove, claims of causality. If you are interested in exploring claims of causality, there are a variety of techniques you can use such as quasi-experimental designs like difference-in-differences, segmented regression and other related techniques.

All of the metrics are proxies for the underlying behaviors of interest: The nine metrics used in this analysis are proxies for various behaviors rather than direct measures. The volume and density metrics measure the frequency and characteristics of interactions on the platforms, but do not measure the underlying cognitive and social processes that shape these interactions. For example, the density of explanations ratio (DER_t) metric measures the number of explanations created relative to the number of questions posted. However, it only considers explanations provided in the form of answers on Stack Overflow. It does not account for explanations shared through other mechanisms such as code comments, design discussions or architectural decision records.

Interdependence between metrics: Several of the metrics are interdependent. For example, AD_t incorporates Q_t; PRD_t and DOR_t share commit denominators. Therefore, it is not safe to treat the nine metrics as independent. The nine statistical tests reported in this analysis should not be interpreted as nine independent confirmations of the observed phenomenon. We used the Bonferroni correction for multiple comparisons in this analysis. This correction is a conservative technique used to control the family-wise error rate. Using this correction, we see that five of the nine tests pass the test for significance, while the remaining four (URB_t, AD_t, DOR_t, DER_t) fail.

Platform dependence: This analysis considers data from only two platforms (Stack Overflow and GitHub). The data presented do not capture development-related interactions that have migrated to other platforms such as private Slack channels, internal wikis, Discord communities and AI chat windows. It is possible that some of the declines we observe in these public traces are the result of a shift away from public platforms to other channels.

Confounding variables: There are a number of factors that may be influencing some of the changes we observe. For example, changes to platform policies such as the policy governing question closure on Stack Overflow and updates to the reputation system may impact developer interactions.

Stack Overflow is also a relatively mature knowledge base at this point; it is possible that the need for this information has decreased over time. Stack Overflow banned discussion of ChatGPT content in December 2022. Shifts in communication channels, the rise of remote work and industry economic trends, such as the recent wave of developer layoffs in 2022 and 2023, also have the potential to impact developer behavior. While the observed changes in Stack Overflow and GitHub behaviors coincided with the availability of AI-powered coding tools, this analysis does not consider these potential confounding variables and thus cannot make claims about the causal relationship between these variables and the observed changes.

Lack of data on private interactions: We do not have data on the frequency or nature of AI-assisted interactions that occur in private. Whether or not developers are interacting with AI coding tools to write code is inferred from the timing of the inflection points in these public data with the milestones related to AI coding tool adoption.

8. Mitigation Strategies

The observed declines in DER_t (δ = −0.33) and DOR_t (δ = −0.38) motivate five evidence-grounded mitigation strategies designed to restore knowledge externalization. Each strategy is linked to specific mechanisms and monitoring metrics from Table 1.

Table 4
Mitigation Strategies — Mechanism Linkage and Monitoring Indicators

Strategy	Practice	Targets	Monitoring Met	Intended Outcome
AI-to-Artifact Workflows	Convert AI outputs into PR notes, doc entries with rationale fields	Artifact Loss; Explanation Compression	DER_t, DOR_t, PRD_t	Increase durable externalization
Rationale-First Reviews	PR template: intent, tradeoffs, validation, AI assistance used	Trust Transfer; Explanation Compression	PRD_t, DOR_t	Restore visible reasoning and scrutiny
Protected Mentorship	Scheduled human–human pairing "explain-back" sessions	Mentorship Displacement	DD_t, TFA_t, URB_t	Preserve interperson learning
Community Contribution Routines	"One reusable artifact per sprint" Q&A, docs, postmortems	Private Substitution; Artifact Loss	Q_t, A_t, DER_t	Sustain shared knowledge commons
Governance Guardrails	AI use policy + CI templates enforcing documentation/review expectations	Norm Drift; Trust Transfer	URB_t, PRD_t, DER_t	Align norms; reduce practice fragmentation

Strategies target observable metric improvements. Monitoring via the same metric suite as reported in Table 2. Evaluation: quarterly measurement using the same extraction pipeline.

AI-to-artifact workflows directly target DER_t and DOR_t by requiring developers to distill AI-assisted solutions into persistent knowledge entries with rationale fields, converting private tool interactions into durable artifacts. Rationale-first reviews restore DD_t and PRD_t by embedding explanation requirements into PR templates: intent, tradeoffs, validation steps, and disclosure of AI assistance used. Given the TFA_t increase (δ = +0.52), protected mentorship — scheduled human–

human pairing and explain-back sessions — is particularly urgent for preserving the interpersonal responsiveness that community traces indicate is declining. Community contribution routines and governance guardrails operationalize these practices at the team and project level, with CI-enforceable documentation requirements targeting DER_t at the automation layer.

9. Conclusion

This paper presents observational evidence of the increased adoption of AI-mediated tools being correlated with changes in the way knowledge is generated, disseminated and retained within a public ecosystem. The Loneliness Framework is a socio-technical process model explaining how the introduction of AI pair programming may influence changes in community engagement, peer-to-peer communication, and externalized knowledge. We define 6 distinct mechanisms, namely private substitution, explanation compression, artifact loss, norm drift, trust transfer and mentorship displacement, each accompanied by discriminant empirical proxies and specific conditions for empirical disconfirmation. We note that empirical support is not uniform across the 6 mechanisms. M1 (private substitution), M2 (explanation compression) and M6 (mentorship displacement) are directly supported by platform-trace evidence. M3 (artifact loss) and M4 (norm drift) have moderate empirical evidence. M5 (trust transfer) remains a theoretical construct that cannot be empirically inferred with the available trace data.

We conduct a fully-computed observational analysis of 84 monthly observations (2018-2024) and find that all 9 community metrics show BH-corrected statistically distinguishable trends ($p < 0.05$), though only 5 remain significant under the more conservative Bonferroni correction. Effect sizes are moderate to large for 6 out of the 9 metrics, with Q_t ($\delta = -0.61$) and TFA_t ($\delta = 0.52$) showing the largest empirical signals. Our analysis of pre-trend baseline values shows that the post-2022 values lie below the extrapolated secular trend between 2018 and May 2021 for most metrics, with the largest deviations occurring in Q_t and PRD_t starting from Q4 2022. Finally, we find a statistically significant association between DER_t and DOR_t ($\rho_s = 0.72$), suggesting a common underlying latent variable reflecting a decline in documentation-focused externalization behavior. While these results are consistent with the mechanisms hypothesized in the Loneliness Framework, the observational design does not support causal claims.

For practitioners, the results suggest that AI-assisted problem-solving should be viewed as a "draft" layer, one that requires externalization in the form of filled-in rationale fields, additional PR comments, or documentation updates to preserve collective knowledge. For managers, rationale-first code review policies and protected mentorship routines are recommended as a direct response to the largest empirical effect (TFA_t). Finally, for tool designers, AI-enabled pair programming tools should allow 1-click export of the AI chat history as a structured PR or issue comment to reduce the cognitive and ergonomic costs of externalization. Future work should pursue quasi-experimental designs to enhance attribution, conduct ethnographic studies to study norm formation in AI-assisted teams, and develop semantically richer proxies for explanation quality beyond volume-based traces.

Data Availability Statement

The datasets used in this study are publicly available. Stack Overflow data were obtained from the Stack Exchange Data Dump (https://archive.org/details/stackexchange). GitHub data were obtained from GH Archive (https://www.gharchive.org). The replication package, including analysis scripts, extracted datasets, and the pre-registered analysis plan, is available at: https://doi.org/10.5281/zenodo.18955743.

Funding
This research received no external funding.

Conflicts of Interest
The author declares no conflict of interest.

References

[1] S. Peng, E. Kalliamvakou, P. Cihon, and M. Demirer, "The Impact of AI on Developer Productivity: Evidence from GitHub Copilot," arXiv:2302.06590, 2023. doi: 10.48550/arXiv.2302.06590.

[2] X. Zhou, P. Liang, B. Zhang, Z. Li, A. Ahmad, M. Shahin, and M. Waseem, “Exploring the problems, their causes and solutions of AI pair programming: A study on GitHub and Stack Overflow,” J. Syst. Softw., vol. 219, Art. no. 112204, 2025. doi: 10.1016/j.jss.2024.112204.

[3] A. Ziegler et al., "Productivity Assessment of Neural Code Completion," in Proc. 44th Int. Conf. Software Engineering: Software Engineering in Practice (ICSE-SEIP), 2022, pp. 23–29. doi: 10.1145/3510454.3517040.

[4] P. Vaithilingam, T. Zhang, and E. L. Glassman, "Expectation vs. Experience: Evaluating code generation tools," in Proc. CHI EA, 2022. doi: 10.1145/3491101.3519665.

[5] S. Abrahao et al., "Software Engineering by and for Humans in an AI Era," ACM TOSEM, vol. 34, no. 2, 2025. doi: 10.1145/3717804.

[6] J. Wang et al., "Software Developers’ Perceptions of AI Code Assistants," IEEE Softw., vol. 40, no. 4, 2023. doi: 10.1109/MS.2023.3267890.

[7] M. Kazemitabaar et al., "Studying the effect of AI-assisted code generation on students’ performance," in Proc. ICER, 2023. doi: 10.1145/3568813.3600139.

[8] M. A. Storey, C. Lee, and K. Foster-Marks, "The New Developer: AI Skill Threat, Identity Change & Developer Thriving," Developer Success Lab, 2024. [Online]. Available: https://dsl.pubpub.org/pub/the-new-dev/release/1.

[9] A. W. Woolley, C. F. Chabris, A. Pentland, N. Hashmi, and T. W. Malone, "Evidence for a collective intelligence factor in the performance of human groups," Science, vol. 330, no. 6004, pp. 686–688, 2010. doi: 10.1126/science.1193147.

[10] T. W. Malone and M. S. Bernstein, Handbook of Collective Intelligence. Cambridge, MA: MIT Press, 2022. doi: 10.7551/mitpress/10834.001.0001.

[11] D. Ha and Y. Tang, "Collective intelligence for deep learning: A survey," Collect. Intell., vol. 1, no. 2, 2022. doi: 10.1177/26339137221114874.

[12] R. Xiao, "Four development stages of collective intelligence," Front. Inf. Technol. Electron. Eng., vol. 25, no. 1, 2024. doi: 10.1631/FITEE.2300459.

[13] J. Surowiecki, The Wisdom of Crowds. New York, NY: Doubleday, 2004.

[14] U. Schmitt, "Designing decentralized knowledge management systems," Kybernetes, vol. 49, no. 10, 2020. doi: 10.1108/K-03-2019-0215.

[15] D. E. Forsythe, "Engineering knowledge: The construction of knowledge in AI," Soc. Stud. Sci., vol. 23, no. 3, 1993. doi: 10.1177/0306312793023003002.

[16] S. Imai, "Is GitHub Copilot a substitute for human pair-programming?" arXiv:2206.15331, 2022. doi: 10.48550/arXiv.2206.15331.

[17] M. Valový, “Psychological aspects of pair programming: A mixed-methods experimental study,” in Proc. 27th Int. Conf. Eval. Assess. Softw. Eng. (EASE), 2023, pp. 210–216. doi: 10.1145/3593434.3593458.

[18] P. Bassner, B. Lenk-Ostendorf, R. Beinstingel, T. Wasner, and S. Krusche, “Less stress, better scores, same learning: The dissociation of performance and learning in AI-supported programming education,” Comput. Educ.: Artif. Intell., vol. 10, Art. no. 100537, 2025. doi: 10.1016/j.caeai.2025.100537.

[19] A. Welter et al., "From Developer Pairs to AI Copilots: A Comparative Study on Knowledge Transfer," arXiv:2506.04785, 2025. doi: 10.48550/arXiv.2506.04785.

[20] B. J. S. Estácio and R. Prikladnicki, "Distributed pair programming: A systematic literature review," Inf. Softw. Technol., vol. 63, 2015. doi: 10.1016/j.infsof.2015.03.001.

[21] G. Fan, D. Liu, R. Zhang, and L. Pan, "The impact of AI-assisted pair programming on student motivation, programming anxiety, collaborative learning, and programming performance: A comparative study with traditional pair programming and individual approaches," Int. J. STEM Educ., vol. 12, Art. no. 16, 2025. doi: 10.1186/s40594-025-00537-3.

[22] F. Song, A. Agarwal, and W. Wen, "The impact of generative AI on collaborative open-source software development," arXiv:2410.02091, 2024. doi: 10.48550/arXiv.2410.02091.

[23] G. Burtch, D. Choi, and K. Kim, "The consequences of generative AI for online knowledge communities," PNAS Nexus, vol. 3, no. 5, 2024. doi: 10.1093/pnasnexus/pgae110.

[24] S. Kabir, D. N. Udo-Imeh, B. Kou, and T. Zhang, "Is Stack Overflow obsolete? An empirical study of the characteristics of ChatGPT answers to Stack Overflow questions," in Proc. CHI Conf. Human Factors Comput. Syst. (CHI), 2024, pp. 1–17. doi: 10.1145/3613904.3642596.

[25] H. Hao et al., "An empirical study on developers' shared conversations with ChatGPT in GitHub pull requests and issues," Empir. Softw. Eng., vol. 29, no. 4, 2024. doi: 10.1007/s10664-024-10540-x.

[26] A. Önden, K. Kara, İ. Önden, G. C. Yalçın, V. Simic, and D. Pamucar, "Exploring the adoption of the metaverse and chat generative pre-trained transformer: A single-valued neutrosophic Dombi Bonferroni-based method for the selection of software development strategies," Eng. Appl. Artif. Intell., vol. 133, Art. no. 108378, 2024. doi: 10.1016/j.engappai.2024.108378.

[27] A. Önden and M. Alnour, "ChatGPT and OpenAI: A comprehensive bibliometric review," J. Soft Comput. Decis. Anal., vol. 1, no. 1, pp. 254–264, 2023. doi: 10.31181/jscda11202324.

[28] L. Da Silva, J. Samhi, and F. Khomh, "LLMs and Stack Overflow discussions: Reliability, impact, and challenges," J. Syst. Softw., vol. 230, Art. no. 112541, 2025. doi: 10.1016/j.jss.2025.112541.

[29] C. Treude, "AI Coding Assistants and the Future of Stack Overflow," IEEE Softw., vol. 41, no. 2, 2024. doi: 10.1109/MS.2024.3399812.

[30] M. Kiygi-Calli, E. Merdin-Uygur, A. Önden, and M. El Oraiby, "Understanding customer conversations in social media support interactions: Divergent sentiments in material and experiential brands," Global Knowl. Mem. Commun., pp. 1–22, 2025. doi: 10.1108/GKMC-02-2025-0098.

[31] A. Bruni, S. Gherardi, and L. L. Parolin, "Knowing in a system of fragmented knowledge," Mind, Cult., Activity, vol. 14, 2007. doi: 10.1080/10749030701307754.

[32] O. Schilke and M. Reimann, "The transparency dilemma: How AI disclosure erodes trust," Organ. Behav. Hum. Decis. Process., vol. 188, Art. no. 104405, 2025. doi: 10.1016/j.obhdp.2025.104405.

[33] T. Dey, S. Mousavi, E. Ponce, T. Fry, B. Vasilescu, A. Filippova, and A. Mockus, "Detecting and characterizing bots that commit code," in Proc. MSR, 2020, pp. 209–219. doi: 10.1145/3379597.3387478.

[34] J. Romano, J. D. Kromrey, J. Coraggio, and J. Skowronek, "Appropriate statistics for ordinal level data: Should we really be using t-test and Cohen's d for evaluating group differences on the NSSE and other surveys?" in Proc. Annu. Meeting Florida Assoc. Institutional Res., 2006, pp. 1–33.