From Teacher to Colleague: How Coding Experience Shapes Developer Perceptions of AI Tools

ILYA ZAKHAROV, JetBrains Research, Serbia
EKATERINA KOSHCHENKO, JetBrains Research, Netherlands
AGNIA SERGEYUK, JetBrains Research, Serbia

AI-assisted development tools promise productivity gains and improved code quality, yet their adoption among developers remains inconsistent. Prior research suggests that professional expertise influences technology adoption, but its role in shaping developers' perceptions of AI tools is unclear. We analyze survey data from 3380 developers to examine how coding experience relates to AI awareness, adoption, and the roles developers assign to AI in their workflow. Our findings reveal that coding experience does not predict AI adoption but significantly influences mental models of AI's role. Experienced developers are more likely to perceive AI as a junior colleague, a content generator, or assign it no role, whereas less experienced developers primarily view AI as a teacher. These insights suggest that AI tools must align with developers' expertise levels to drive meaningful adoption.

Additional Key Words and Phrases: Human-Computer Interaction, Artificial Intelligence, Software Engineering, User Studies, User Experience

ACM Reference Format:
Ilya Zakharov, Ekaterina Koshchenko, and Agnia Sergeyuk. 2025. From Teacher to Colleague: How Coding Experience Shapes Developer Perceptions of AI Tools. 1, 1 (February 2025), 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

AI-assisted development tools are increasingly being integrated into professional coding environments, promising productivity gains, code quality improvements, and automation of routine tasks [7, 9, 12]. Despite these advancements, AI adoption varies widely among developers, with some readily integrating AI-driven tools into their workflows while others remain skeptical or resistant. Prior research suggests that personal experience and technical knowledge shape users' mental models, which in turn influence their perception of a tool's usability and affect adoption [2], suggesting that the more experienced professionals are, the more they are prone to the adoption of the new tools in the field. However, it is unclear whether this trend holds in software engineering and what "roles" — specific functions AI can perform within a developer's workflow, representing mental models of a tool — are associated with its adoption.

Our research seeks to understand the connection between professional expertise, in our case in coding, and individuals' perspectives on AI. Overall, our research questions may be formulated as follows:

- **RQ1**: *How is coding experience related to the adoption of AI tools?*

Authors' addresses: Ilya Zakharov, ilia.zaharov@jetbrains.com, JetBrains Research, Belgrade, Serbia; Ekaterina Koshchenko, ekaterina.koshchenko@jetbrains.com, JetBrains Research, Amsterdam, Netherlands; Agnia Sergeyuk, agnia.sergeyuk@jetbrains.com, JetBrains Research, Belgrade, Serbia.

Manuscript submitted to ACM

- **RQ2**: *How do coding experience and exposure to AI tools relate to the "roles" that developers can assign for AI in their workflow?*

In analyzing survey data from 3380 developers conducted in Q2 2024 [8], we found that developers' experience does not predict whether they adopt or are even aware of AI tools. However, experience does shape the roles they assign to these tools. Specifically, more experienced developers are more likely to perceive AI as either a junior colleague, a content generator, or assign it no role at all. In contrast, less experienced developers tend to describe AI in a teacher role. These findings suggest that senior developers may not perceive enough value in AI tooling to integrate it into their workflow, whereas novice programmers may rely on AI tools for guidance and learning. This highlights the importance of aligning AI tools with users' mental models — especially for developers with varying levels of expertise — by designing features that address their distinct expectations and needs.

2 RELATED WORK

The relationship between professional expertise and the adoption of AI tools in software development is complex. Drawing on established models of technology acceptance, prior research indicates that external variables — such as organizational context and prior experience — play a pivotal role in shaping attitudes toward new technologies [3, 13].

Prior research suggests that personal experience and technical knowledge shape users' mental models, which in turn influence their perception of a tool's usability and affect adoption [2]. A recent study on how developers perceive and interact with AI-driven code completion suggests that discrepancies between a developer's mental model and the tool's operations can lead to resistance to its adoption [4]. Mental model mismatches have been shown to result in frustration, reduced trust, and eventual abandonment of AI tools [5].

Based on Ericsson and Smith [6] general theory of expertise, it might be suggested that experts tend to rely on years of deliberate practice and accumulated domain knowledge, thereby requiring robust evidence of possible benefits before adopting AI systems. In contrast, novices are often more open to experimentation and may rapidly complete tasks with AI tools. However, they tend to over-rely on it, potentially veering away from correct solutions, lacking the expertise to catch subtle errors [10], sometimes relying on surface-level cues of the tool functionality [14].

Prior research offers mixed findings on how expertise influences AI adoption, making its impact on software developers unclear. Moreover, while mental models shape adoption, little is known about how experience levels affect the "roles" developers assign to AI tools. We think that understanding this interplay is key to designing AI tools that align with developers' needs and encourage broader adoption.

3 METHOD

To study the relationship between coding experience and AI tool adoption, as well as how developers' mental models shape the "roles" they assign to AI within their workflow, we analyzed the subset of data from JetBrains' State of Developer Ecosystem Report 2024 [8]. This report is designed to track developers' habits and attitudes towards various programming languages, applications, tools, frameworks, and even developers' lifestyles and habits. The main survey was conducted in May–June 2024, and the main report is based on the input of 23,262 developers from 171 countries and regions. We used a subset of the whole survey that relates to the study's research questions. The overall methodology for the survey can be found through the link [1].

For the purposes of the present research, we derived and analyzed answers to the following questions regarding:

[1] JetBrains' State of Developer Ecosystem Report Methodology — https://www.jetbrains.com/lp/devecosystem-2024/methodology/

- Participants' coding experience in years;
- The total number of AI tools for development people tried ("AI experience") and heard about ("AI awareness");
- Roles of AI tools for developers.

The roles suggested in the survey were based on prior studies in the field [11] and previous surveys conducted by JetBrains [2]. The available options included: assistant, tool, reference guide, content generator, problem solver, collaborator, senior colleague, junior colleague, teacher, advisor, reviewer, copilot, and companion. Additionally, there was an option to specify your own role or select "None of the above".

The acquired subset was cleaned to focus on developers (people who indicated "Coding/Programming" as a part of their work activities), reducing the initial sample to 19820 participants. From this sample, 8073 participants answered the questions related to their experience with AI, and 3380 participants answered chose at least one option for the question "In which roles do you see AI tools for developers".

To understand the relationship between coding experience and adoption of AI tools, we calculated Spearman's correlations between coding experience, "AI experience," and "AI awareness" metrics. To examine the association between coding experience and the likelihood of a respondent selecting a given role, we fitted a series of logistic regression models, estimated using the maximum likelihood method. The estimated coefficient for coding experience was exponentiated to yield an odds ratio, which quantifies the change in odds of selecting a given role for each additional unit of coding experience. P-values were adjusted for the multiple comparisons using Benjamini & Hochberg false discovery rate procedure [1].

4 RESULTS

Variables	Spearman's ρ	p-value	N
AI Experience vs. AI Awareness	0.546	< 0.001	8073
AI Awareness vs. Coding Experience	0.031	0.005	8073
AI Experience vs. Coding Experience	-0.005	0.652	8073

Table 1. Spearman Correlation Results Among AI Awareness, AI Experience, and Coding Experience
Significant correlations are shown in bold.

We first examined the relationship between Coding Experience and both Awareness of and Experience with AI tools for coding. As shown in Table 1, AI Awareness and AI Experience are significantly correlated, as expected. However, there is no correlation between Coding Experience and AI Experience, and the correlation between Coding Experience and AI Awareness, while statistically significant, is negligible (r=0.031).

Next, we examined whether Coding Experience is associated with the specific "roles" users assign to AI tools. Logistic regression results (Table 2) show that four response types are significantly linked to Coding Experience: three specific roles — "teacher", "junior colleague", and "content generator" — as well as the "None of the above" option. The odds ratios indicate that developers with more coding experience are more likely to choose "junior colleague", "content generator", or "None of the above", while those with less experience more often select "teacher".

[2] JetBrains' State of Developer Ecosystem Reports — https://blog.jetbrains.com/category/deveco/

Role	Coefficient	Std. Err.	z	P>|z|	Odds Ratio	CI Lower	CI Upper	p-adj
None of the above	0.322	0.080	4.036	0.000	1.380	1.180	1.614	0.001
As a teacher	-0.124	0.042	-2.986	0.003	0.883	0.814	0.958	0.017
As a junior colleague	0.157	0.053	2.935	0.003	1.170	1.053	1.299	0.017
As a content generator	0.083	0.030	2.780	0.005	1.087	1.025	1.152	0.020
As a senior colleague	-0.134	0.074	-1.803	0.071	0.875	0.757	1.012	0.214
As a problem solver	-0.034	0.029	-1.163	0.245	0.967	0.913	1.023	0.513
As a copilot	0.027	0.024	1.095	0.274	1.027	0.979	1.078	0.513
As an advisor	0.030	0.029	1.017	0.309	1.030	0.973	1.092	0.515
As a reference guide	0.014	0.023	0.610	0.542	1.014	0.970	1.060	0.812
As a tool	-0.009	0.019	-0.459	0.646	0.991	0.956	1.028	0.875
As a collaborator	0.016	0.041	0.386	0.700	1.016	0.937	1.101	0.875
As an assistant	-0.004	0.019	-0.207	0.836	0.996	0.960	1.034	0.919
As a reviewer	0.005	0.030	0.180	0.857	1.005	0.948	1.066	0.919
As a companion	-0.003	0.037	-0.070	0.944	0.997	0.928	1.072	0.944
Other, please specify:	0.176	0.111	1.578	0.114	1.192	0.958	1.483	0.286

Table 2. Logistic regression results for AI roles. The “p-adj” represents the FDR-adjusted p-value. Significant associations are shown in bold.

5 DISCUSSION

The goal of our study was to investigate the relationship between Coding Experience, Awareness of and Experience with AI tools for development, and perceptions of them.

Although the correlation analysis indicated a statistically significant relationship between coding experience and AI Awareness, the extremely low correlation coefficient (r = 0.031) suggests that coding experience alone accounts for only a negligible portion of the variance in AI Awareness. This finding implies that other factors —such as exposure to technology, organizational culture, or individual curiosity — might be more influential in shaping awareness of AI tools.

The logistic regression analysis demonstrates that coding experience is associated with distinct “roles” that users assign to AI tools. Participants with higher levels of coding experience tend to perceive the AI tool as a junior colleague or content generator. Conversely, less experienced coders tended to view the AI tool as a teacher. This divergence may reflect underlying differences in expectations: novice programmers might seek instructional support and guidance from AI systems, whereas more seasoned developers may be looking for a collaborative but not necessarily reliable partner that can help with content generation and problem solving.

Notably, developers with more coding experience also chose the “None of the above” option significantly more often. This may suggest that experienced professionals might be relatively rigid or skeptical toward new emerging technologies and that a considerable part of that community currently tends not to see any useful “role” for AI tools for coding. This observation is particularly interesting given the negligible association between coding expertise and AI Awareness and the association between expertise and AI Experience. In other words, individuals with greater coding expertise do not seem to demonstrate a higher interest in experimenting with AI tools for coding.

To sum up, our findings indicate that while coding experience minimally impacts the adoption of AI tools, it significantly shapes developers’ perceptions of the “role” of AI tooling in their workflow and their willingness to engage with such tools.

ACKNOWLEDGMENTS

We thank the JetBrains Market Research & Analytics Team, especially the Surveys Team, for designing and conducting the State of Developer Ecosystem Survey and for their support throughout our study.

DATA AND CODE AVAILABILITY STATEMENT

The JetBrains' State of Developer Ecosystem Report 2024 is available at [8]. The code for data cleaning and analysis is available upon request.

REFERENCES

[1] Yoav Benjamini and Yosef Hochberg. 1995. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *Journal of the Royal Statistical Society: Series B (Methodological)* 57 (1995). https://doi.org/10.1111/j.2517-6161.1995.tb02031.x

[2] Mei-Ling Chen and Hao-Chuan Wang. 2018. How personal experience and technical knowledge affect using conversational agents. In *Companion Proceedings of the 23rd International Conference on Intelligent User Interfaces*. 1–2.

[3] Fred D Davis. 1989. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS quarterly* (1989), 319–340.

[4] Giuseppe Desolda, Andrea Esposito, Francesco Greco, Cesare Tucci, Paolo Buono, and Antonio Piccinno. 2025. Understanding User Mental Models in AI-Driven Code Completion Tools: Insights from an Elicitation Study. *arXiv preprint arXiv:2502.02194* (2025).

[5] Shehzaad Dhuliawala, Vilém Zouhar, Mennatallah El-Assady, and Mrinmaya Sachan. 2023. A Diachronic Perspective on User Trust in AI under Uncertainty. *arXiv preprint arXiv:2310.13544* (2023).

[6] K Anders Ericsson and Jacqui Smith. 2002. *Prospects and limits of the empirical study of expertise: An introduction.* MIT Press.

[7] Saki Imai. 2022. Is github copilot a substitute for human pair-programming? an empirical study. In *Proceedings of the ACM/IEEE 44th International Conference on Software Engineering: Companion Proceedings*. 319–321.

[8] JetBrains. 2024. The State of Developer Ecosystem 2024. https://www.jetbrains.com/lp/devecosystem-2024/ Accessed: February 13, 2025.

[9] Sida Peng, Eirini Kalliamvakou, Peter Cihon, and Mert Demirer. 2023. The impact of ai on developer productivity: Evidence from github copilot. *arXiv preprint arXiv:2302.06590* (2023).

[10] James Prather, Brent N Reeves, Paul Denny, Brett A Becker, Juho Leinonen, Andrew Luxton-Reilly, Garrett Powell, James Finnie-Ansley, and Eddie Antonio Santos. 2023. "It's Weird That it Knows What I Want": Usability and Interactions with Copilot for Novice Programmers. *ACM Transactions on Computer-Human Interaction* 31, 1 (2023), 1–31.

[11] Steven I Ross, Fernando Martinez, Stephanie Houde, Michael Muller, and Justin D Weisz. 2023. The programmer's assistant: Conversational interaction with a large language model for software development. In *Proceedings of the 28th International Conference on Intelligent User Interfaces*. 491–514.

[12] Mircea-Serban Vasiliniuc and Adrian Groza. 2023. Case study: using AI-assisted code generation in mobile teams. In *2023 IEEE 19th International Conference on Intelligent Computer Communication and Processing (ICCP)*. IEEE, 339–346.

[13] Viswanath Venkatesh, Michael G Morris, Gordon B Davis, and Fred D Davis. 2003. User acceptance of information technology: Toward a unified view. *MIS quarterly* (2003), 425–478.

[14] Aaron S Yarlas and Vladimir M Sloutsky. 2000. Problem representation in experts and novices: Part 1. Differences in the content of representation. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, Vol. 22.