Silent ROS Community: Let's Start Discussing the Impact of AI Adoption on Public Knowledge Sharing

Michel Albonico
MMMI TEK, Center for Software Technology (CST), University of Southern Denmark
Vejle, Denmark
mical@mmmi.sdu.dk

Douglas Ferreira de Souza
Programa de Pós-Graduação em Computação Aplicada (PPGCA), Universidade Tecnológica Federal do Paraná
Curitiba, Brasil
douglassouza.src@gmail.com

Adolfo Neto
Departamento Acadêmico de Informática (DAINF), Universidade Tecnológica Federal do Paraná
Curitiba, Brasil
adolfo@utfpr.edu.br

Abstract

Generative Artificial Intelligence tools are increasingly changing how developers solve technical problems, potentially shifting knowledge sharing from public forums to private interactions with Artificial Intelligence (AI) assistants. While this may improve productivity, it may also reduce the amount of publicly available knowledge in software communities. In this paper, we start investigating how AI adoption may be associated with the decline in public discussions among software developer communities, focusing on a previously active community: the Robotic Operating System (ROS) ecosystem. To estimate periods of AI-assisted development, we analyze popular ROS projects on GitHub and compare them with the evolution of the number of discussions across ROS-related communication channels. We then discuss the potential implications on the rise of AI is contributing to the emergence of "silent communities", making technical knowledge increasingly private and less accessible to future developers and researchers.

Keywords

AI, ROS, Software Engineering, Q&A

1 Introduction

The debate surrounding the rise of Artificial Intelligence (AI) in software engineering[1] has largely focused on issues such as ethics and privacy [7], authorship [11], intellectual property [15], bias [4], and the potential replacement of human labor [5]. While these topics are undeniably important, a less visible although still important consequence of AI adoption remains underexplored: the gradual disappearance of public technical discussions on question and answer (Q&A) platforms.

Software communities have relied on public forums such as Stack Overflow to ask questions, report issues, discuss workarounds, and collectively build knowledge [8]. Those platforms have played a central role not only in the growth and maintenance of software ecosystems [9], but for the research community to understand developers behavior and challenges [1] and improve software documentation [14]. However, this scenario has been shifting, mainly after the recent widespread adoption of AI tools based on large language models (LLMs) [6].

With the use of AI tools, such as ChatGPT, GitHub Copilot, and Claude, developers tend to get immediate answers to their questions, which accelerates the process, while limits the public exposing of their issues. From the perspective of the individual developer, this shift may be beneficial. AI tools can reduce waiting time, provide explanations in natural language, generate code snippets, and help developers quickly overcome obstacles. However, at the ecosystem level, this transition may produce negative impacts. If fewer developers ask questions publicly, fewer discussions are archived, fewer solutions become reusable, and less knowledge becomes available to future developers, or even to further train the LLM models.

The decline in activity on public discussion platforms may be particularly concerning in niche and highly technical ecosystems such as the Robot Operating System (ROS). ROS is one of the most prominent frameworks for robotics software development, comprising thousands of reusable open-source packages [12], and relies heavily on an active community for its subsistence. Within this ecosystem, public discussion channels have historically played a fundamental role in supporting developers, facilitating problem-solving, and enabling knowledge sharing [10]. Unlike general-purpose software communities, which have largely relied on Stack Overflow, the ROS community traditionally centralized discussions around ROS Answers until 2023, and after that, on Robotics Stack Exchange (hosted by Stack Overflow). This shift raises the first important **research point (1)**: the extent of which ROS community, similarly to other software ecosystems, reduced discussions on its public Q&A platforms.

The possible reduction on the number of public questions and discussions raises multiple concerns about the sustainability of software communities [8], including the ROS one: (i) researchers may lose important sources of empirical evidence about developer pain points, and reduce their work on evidencing community software development aspects; (ii) project maintainers may become less aware of recurring problems in their software; (iii) documentation decay may become more difficult to address; and (iv) in the long term, knowledge may become increasingly concentrated in private interactions between developers and AI systems, making it inaccessible to the wider community. At this point, these are only speculative hypothesis, which require a extensive investigation.

In **this paper**, we (i) investigate whether ROS discussion channels have experienced a decline in activity, similar to trends observed in Stack Overflow; (ii) analyze how the adoption of AI tools may be associated with fluctuations in public community discussions over time; and (iii) discuss the potential implications of these trends for the sustainability of the ROS community. More specifically, we:

[1]A more general overview of the impacts—some of them negative—of AI on society can be found in [2].

• mine all posts from the main ROS discussion platforms, collecting metadata such as date, time, title, and communication channel. This dataset enables us to characterize temporal trends in issue-related discussions within the ROS community.
• highlight the first evidences of AI usage in widely adopted ROS-based projects hosted on GitHub. These projects are identified through the ROS Index, where core ROS packages are cataloged, as well as through a complementary manual selection of popular repositories, ranked by indicators such as stars, watchers, and forks, that are not listed in the ROS Index.
• mine commits, issues, and pull requests data and metadata from a representative set of public ROS projects. From commits, we extract signals that may indicate the use of AI-assisted development, such as the presence of AI-related keywords in their titles, characteristics of modified files and change volumes, author-related metadata, and configuration artifacts suggesting the integration of AI-based tools. Looking at the mined data we want to evidence another **research point (2)** on whether there is a possible relationship between AI adoption in ROS projects and the decrease of public discussion activity.

This study mainly contributes to an emerging debate about the hidden consequences of AI adoption in software engineering communities. First, it introduces the notion of "silent communities", where supposedly developers increasingly solve problems privately through AI rather than publicly through community interactions. Second, it provides an empirical analysis of AI adoption indicators and discussion decline in the ROS ecosystem. Finally, it contributes a longitudinal dataset of ROS packages and public discussions that may be valuable for future research.

2 Robot Operating System (ROS) Ecosystem

Robot Operating System (ROS) is an open-source middleware framework widely adopted in robotics research and industry to support the development of modular and distributed robotic systems. ROS provides a collection of tools, libraries, communication layer, and conventions that simplify the integration of sensors, actuators, control algorithms, and software components into robotic applications. Its architecture is based on nodes that communicate through topics, services, and actions, enabling loose coupling and reusability of software components. Over the years, ROS has become the de facto standard for robotics software development [10], with an extensive ecosystem, strong community support, and compatibility with simulation platforms.

ROS consists of a collection of open-source packages, which are all listed at ROS Index[2]. By indexing packages hosted primarily on GitHub, ROS Index facilitates software reuse, ecosystem exploration, and empirical studies involving ROS projects. It can be used by researchers and practitioners to identify relevant packages for robotic applications.

ROS community has historically relied on dedicated forums for discussing technical issues and troubleshooting problems. Until 2023, ROS Answers[3] has been the main question-and-answer platforms used by robotics developers. More recently, with the discontinuation of ROS Answers, much of the ROS community activity has migrated to Robotics Stack Exchange.

[2]https://index.ros.org/

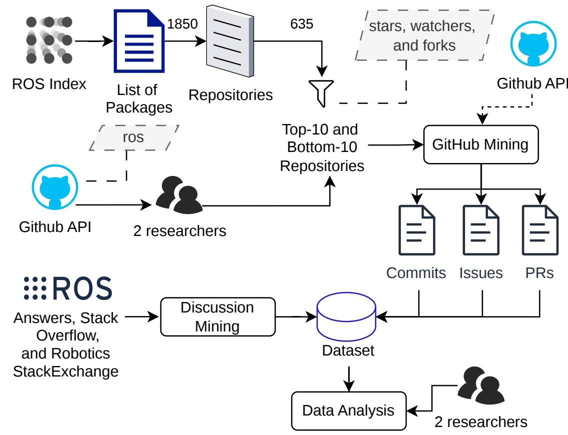


Figure 1: Mining repositories and data analysis workflow.

3 Study Design

Figure 1 illustrates the workflow of the approach, divided in three main phases: i) GitHub mining, ii) discussion mining, and iii) data analysis, the details of which are presented in the remainder of this section.

Before the mining phase, we preselect a representative list of ROS-based repositories, which could also be considered as a pre-mining or data preparation phase. For our dataset to be representative, we look for two different sources: the ROS Index, where all the ROS 2 dependencies are listed, and other GitHub hosted repositories, which are popular and are not listed in the ROS Index.
• **ROS Index** lists 1,815 packages for the latest ROS distribution (i.e., Kilted). We crawl the ROS Index website using a Selenium-based approach and store the retrieved data in a comma-separated values (CSV) file. Next, we group packages by repository and analyze the corresponding repositories via the GitHub API, ranking them by number of stars (a widely adopted proxy for popularity in the mining software repositories). We also make sure that only the repositories updated within the last 6 months are selected, aiming at active development. We then select the top-10 repositories (most popular) and the bottom-10 (least popular). This selection enables us to compare patterns across repositories with different levels of popularity and, potentially, different collaboration dynamics. Our analysis later confirms that bottom-10 repositories exhibit substantially fewer commits, issues, and pull requests. We only consider 20 repositories considering they already resulted in 93,239 commits, which are multi-dimensionally mined. Therefore, a longer list of repositories would compromise the feasibility of the analysis.
• **Other ROS repositories** are abundant on GitHub. To identify them, we queried the GitHub API using the `ros` keyword, applying

[3]https://answers.ros.org/

the same filtering criteria adopted for ROS Index repositories, which yielded 532 candidate repositories. Subsequently, two researchers manually reviewed the repositories, meeting iteratively to reach consensus. Each repository was classified according to whether it is ROS-related and whether it supports the current ROS version (i.e., ROS 2). In total, we identified 484 ROS repositories, of which 184 are ROS 2–based. Finally, we ranked the repositories by number of stars and selected the top-10 and bottom-10 for further analysis. This delivers a complementary dataset in which we expect less structured and formalized collaboration among contributors, as these projects are not explicitly curated or intended for broad reuse, unlike those listed in the ROS Index.

3.1 GitHub Mining

We use the GitHub API to mine all the *commits*, *issues*, and *pull requests* metadata from 40 selected projects. Each set of metadata is stored into separated CSV files for further analysis.

For the commits, we mine the commit title, message, creation date, modified files, number of lines added and deleted, and aggregate change statistics (total and average), as well as AI-related evidence signals. We exclude files that are entirely new or completely removed, as such changes do not provide meaningful evidence regarding AI-assisted development. Furthermore, we restrict the mining to source code files, identified a comprehensive list of file extensions spanning multiple programming languages.

The identification of AI-assisted commits is based on a scoring mechanism that aggregates multiple observable signals extracted from repository data. For each commit, an *AI score* is computed by analyzing: (i) the commit message, (ii) the set of modified files, and (iii) author-related metadata retrieved via the GitHub API. First, the approach searches for predefined keywords associated with AI tools (e.g., "ChatGPT", "Copilot", "Claude") in commit messages, assigning a weight of +5, as these may explicitly indicate AI-assisted contributions. Second, if the author's login contains bot-related keywords (e.g., "copilot", "claude"), a weight of +4 is assigned. Additionally, if the author type is labeled as *Bot*, a further +2 is added, reflecting automated contributions such as those generated by GitHub Actions. Third, the presence of file paths or configuration artifacts indicative of AI tooling (e.g., `copilot.yml`, `.cursor/`) contributes an additional +3 to the score, capturing cases where the repository is explicitly configured to support AI-assisted development. We deliberately avoid analyzing the contents of source code (e.g., code patterns or comments), which is a matter for a future and deeper study.

A commit is classified as AI-assisted if its computed score is greater than zero, while the magnitude of the score serves as a proxy for confidence. In addition to this classification, the method records explanatory metadata, including the specific signals that triggered detection and the inferred AI tools involved. This design provides both quantitative and qualitative insights into AI usage within software repositories.

3.2 Discussion Mining

For ROS Discussion, we crawled ROS Answers and StackOverflow, two popular channels where ROS developers and users discuss their main issues, identifying each of them with the date an time they were posted. For ROS Answers, we crawl all the discussion, while for Stack Overflow, we use the keyword `ros` to identify discussion topic titles. Since ROS Answers have been migrated to Robotics StackExchange, we also mined all the discussion there.

3.3 Data Analyis

The goal of this analysis is not to definitively prove that a repository employs AI-based tools, but rather to estimate periods during which AI-assisted development becomes more prevalent. We rely on a quantitative analysis that focus on identifying trends in discussion activity over time. In this phase of the research, we analyze generated graphs and re-visit relevant data points and websites trying to reason about the visual findings. We also rely on a linear regression analysis on the discussion messages to evaluate whether the number of discussions changed systematically over time. We model the relationship between monthly discussion counts and time, which helps us to identifying the direction and strength of the temporal trend.

4 Results

In this section, we illustrate and describe project activity over time using a representative subset of projects, along with indicators related to AI adoption and usage. Finally, we illustrate and analyzed the evolution of ROS-related discussions over time.

4.1 AI Usage Indicators

Figure 2 illustrates a subset of the graphs generated from *commits* potentially associated with AI-assisted development. Since our analysis is currently limited to commit metadata, such as commit messages and authorship information, and does not yet include codebase-level inspection for AI traces, the results cannot be considered conclusive. Nevertheless, this preliminary analysis suggests a continuous presence of AI-assisted contributions since approximately 2021, with a more pronounced increase from 2023 onward. Among the most popular projects at the top of the ROS Index, we observed a lower proportion of AI-related commits when compared to the remaining projects. This may indicate that the core ROS 2 projects are still being developed primarily through carefully supervised human contributions. Such a pattern is understandable, as these projects are generally maintained by a relatively small group of highly experienced collaborators with deep knowledge of the ROS ecosystem. However, it is still important to *survey the collaborators and maintainers, to understand how seriously they are regarding AI-generated code to be merged to their projects*.

The bottom-10 projects in the ROS Index, as well as the additional ROS-related open-source projects analyzed, exhibited a more expressive presence of AI-associated commits. This behavior is also expected, since these projects tend to be more community-driven and accessible to external contributors, potentially enabling less experienced developers to participate (hypothetically empowered by the support of AI tools). This interpretation is further aligned with the usage patterns observed in ROS Answers, where novice ROS users tend to be more active in seeking assistance.

We also analyzed the amount of code modified per commit as a possible indicator associated with AI-assisted development. Our initial hypothesis was that developers relying on AI tools would introduce larger code modifications, potentially causing GitHub to classify entire files as newly added or extensively rewritten. However, the results did not reveal significant differences during the periods in which AI usage was expected to increase. These findings suggest that *additional analyses are required, particularly static analysis techniques capable of identifying AI-generated code characteristics more directly*. Another possible explanation is that pull requests in these projects undergo strict review and quality control processes, preventing excessively large contributions from being merged. To understand this phenomenon, *a deeper qualitative investigation of merged and rejected pull requests is still necessary.*

4.2 Collaborative Activity Over Time

We also analyzed *pull requests* (PRs) and *issues*; however, the corresponding graphs and detailed data are provided only in the replication package due to space limitations. Mining large GitHub repositories is a time-consuming process, with each of most popular repository requiring more than two hours to be fully processed because of GitHub API restrictions. As a consequence, the current analysis covers only a partial dataset, including five repositories from those previously selected for the *commit* and AI analyses. Therefore, the results should still be considered preliminary.

The initial findings indicate a substantial increase in the number of pull requests after 2021, which is likely associated with the growing popularity of ROS 2 and the expansion of its package ecosystem. From 2023 onward, this increase becomes even more pronounced. One possible explanation is the consolidation of ROS 2 as the main robotics middleware platform, which encouraged a larger number of developers to contribute to open-source ROS projects. Another plausible explanation is the official end-of-life of ROS 1 in 2023, which forced many projects to migrate their codebases to ROS 2, consequently generating a surge of pull requests and maintenance activities. A similar trend can be observed in the number of opened and closed *issues*. Nevertheless, *further qualitative analysis is still necessary to determine whether these changes are also associated with the adoption of AI-assisted development tools*.

Other metrics suggest phenomena that may be more directly associated with the increasing use of AI in software development, particularly the average time required to merge a *pull request* and the number of distinct authors contributing through pull requests. The average pull request merge time nearly doubled after 2023, which may be partially explained by the significant growth in the number of submitted pull requests, increasing the workload for maintainers and reviewers. At the same time, the number of distinct pull request authors also increased considerably during this period. This growth may indicate a lower barrier to contribution, potentially enabled by AI-based coding assistants that facilitate code generation, documentation, and bug fixing, allowing less experienced contributors to participate more actively in ROS open-source projects. We also plan to *further investigate the causes by a future qualitative analysis*.

4.3 ROS Discussion Over Time

Figure 3 illustrates a trend in the ROS Answers discussions that is similar to the decline observed in other developer discussion platforms, such as Stack Overflow.[4] At the beginning of the timeline, the ROS community exhibited an intense discussion, which is expected during the adoption of the ROS ecosystem. Starting around June 2019, however, the volume of discussions began to decrease, with a more pronounced decline between 2020 and early 2021. This reduction may be associated with the maturation of the ROS ecosystem, the availability of more stable documentation and tutorials, and the migration of discussions to alternative communication channels such as Discord and Stack Overflow (whose relevance increased during this period). However, to conclude about that, it is necessary a *further qualitative investigation, which we pretend to complete as a future work*.

In August 2023, the figure shows an abrupt increase in the number of discussions, suggesting the occurrence of a significant community event. This is the exact time ROS Answers was completely migrated to ROS StackExchange[5]. In the ROS Answers webpage[6], now there is only archived discussions, with the last one dating from August $11th$, 2023. In the ROS Discussion channel, in May 2023, this intent had already been announced, where one of the ROS maintainers highlighted the expected benefits of the migration, particularly the improved user experience provided by the Stack Exchange infrastructure, including better moderation support, automated handling of spam and stale questions, and greater visibility through integration with a broader technical community. At the same time, the maintainers also acknowledged the intention to migrate as much discussion content as possible from ROS Answers to Stack Exchange. Since our dataset includes discussions from both ROS Answers and ROS Stack Exchange, the spike observed after 2023 is primarily associated with the reassignment of post creation timestamps during the migration process.

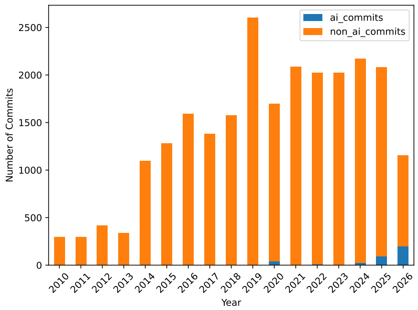
The peak in the graph does seems not to represent a genuine increase in community activity, which remained relatively low over the analyzed period. However, during this period there was also the end of the support for all the ROS 1 distributions, Melodic (in June 2023) and Noetic (in May 2025). This then requires *further qualitative analysis we plan to do as future work to understand whether it is exclusively due to timestamp issues*. Finally, by September 2025, the discussion volume had returned to levels comparable to those observed before the 2023 peak, reinforcing the interpretation that the surge was temporary rather than indicative of sustained long-term growth.

The linear regression analysis confirms the visual observations, with a statistically significant negative trend ($\beta = -1.72$, $t = -16.15$, $p < 0.001$), indicating a consistent decrease in the number of discussions throughout the analyzed period.

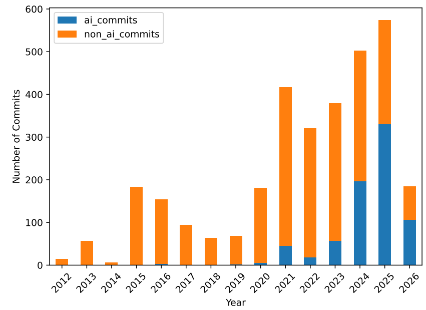
5 Discussion and Future Directions

The results presented in this study provide preliminary evidence that public technical discussions within the ROS ecosystem have

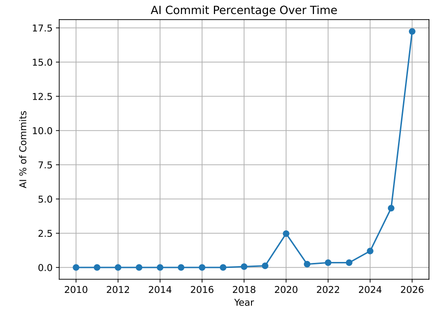
[4]https://www.webdesignerdepot.com/the-decline-of-stack-overflow-where-are-developers-headed-next/
[5]https://robotics.stackexchange.com/
[6]https://answers.ros.org/questions/



(a) AI vs non-AI commits over all the analyzed repositories.



(b) AI vs non-AI commits over the bottom-10 projects on ROS Index.



(c) Trend on the usage of AI for all the analyzed repositories.

Figure 2: Tendency on the use of AI for commits on GitHub-based ROS repositories.

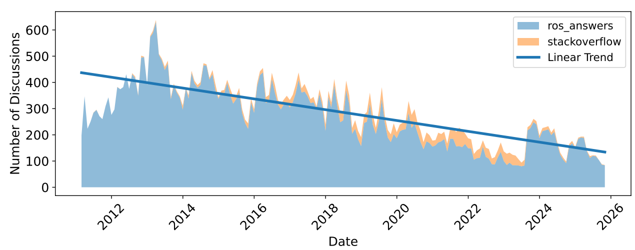


Figure 3: Number of new discussion over the years.

been decreasing over time, particularly after the widespread adoption of AI-assisted development tools. Although the current analysis does not allow us to establish causality between these phenomena, the temporal coincidence between the reduction of discussions and the increase of AI-related development signals suggests that an important transition may be occurring in the way developers seek and share technical knowledge.

One possible interpretation is that developers are progressively replacing public interactions with private conversations mediated by Large Language Models (LLMs), which provide immediate answers, reducing the need to open public questions in discussion forums. From the perspective of individual productivity, this transition may be beneficial, since developers can solve problems faster and with lower communication overhead. However, from the perspective of the software ecosystem, this behavior may produce important long-term consequences.

Public questions, answers, workarounds, and discussions constitute an important repository of empirical evidence regarding developer pain points, recurring bugs, documentation deficiencies, API limitations, and maintenance challenges. A reduction in the volume of these discussions may therefore decrease the visibility of real-world problems faced by developers. This phenomenon may be particularly problematic for ROS because the niche ecosystem heavily depends on collaborative open-source development.

Another important concern involves the sustainability of AI models themselves. Modern LLMs heavily rely on publicly available technical content for their foundation models. If public technical discussions continue to decline, future AI systems may eventually face a reduction in the availability of training data.

Our preliminary findings indicate that AI-related commits appear more frequently in community-driven ROS projects than in core ROS repositories. This may suggest that AI-based tools are lowering the barrier for contribution, enabling less experienced developers to participate more actively in the ecosystem. While this democratization may increase collaboration, it also raises concerns regarding code quality, maintainability, and review processes. Since ROS systems are often used in safety-critical robotic applications, insufficiently reviewed AI-generated code may introduce defects, architectural inconsistencies, or security risks.

As this is an emerging ideas paper, the results are still be considered preliminary. Nevertheless, we believe the findings are sufficiently relevant to motivate broader discussion within the ROS and software engineering communities. Understanding how AI is changing collaborative knowledge sharing practices may become essential for preserving healthy and sustainable open-source software ecosystems in the long term.

5.1 Next Phases of the Study and Methodology

The current study represents an initial step toward understanding the relationship between AI adoption and the reduction of public technical discussions in the ROS ecosystem. Several open questions remain unanswered, requiring additional quantitative and qualitative investigations. In the next phases of the study, we plan to extend both the dataset and the methodological procedures in order to provide stronger evidence regarding the observed phenomena.

Expansion of the analyzed dataset: As a next step, we intend to consider all the repositories we mined from ROS index and GitHub. We also plan to classify the projects according their nature, type of robot, if their are foundational or not, and other aspects that can help understanding how different groups relate to the use of AI.

Improved detection of AI-assisted development: The current strategy for identifying AI usage does not directly inspect the source code itself. As future work, we plan to extend the methodology with static code analysis techniques capable of identifying characteristics commonly associated with AI-generated code. Potential strategies include: i) detection of repetitive coding patterns; ii) identification of AI-generated comments or documentation; and iii) machine learning classifiers trained on an existing labeled dataset [3] of AI-generated and human-written code.

We also intend to analyze pull request discussions and code review comments to investigate whether maintainers explicitly identify or discuss AI-generated contributions during the review process. This is important to understand if the merging is carefully reviewed.

Qualitative analysis of discussions: The quantitative reduction in public discussions does not necessarily imply a complete reduction in collaboration. Developers may have migrated to alternative communication channels such as Discord, Slack, GitHub Discussions, or private organizational platforms. Therefore, we plan to perform a qualitative analysis of discussion content and communication channels. This analysis will investigate: i) which categories of questions are disappearing faster; ii) whether elementary questions regarding debugging and configuration are still present; iii) whether architectural discussions remain public; and iv) whether AI-generated answers are being reposted in public forums.

Survey and interviews with ROS developers: We plan to conduct surveys and semi-structured interviews with contributors and maintainers of ROS projects to confirm the AI-based tools they use to support their projects. The survey will specifically investigate: i) frequency of AI tool usage; ii) types of tasks supported by AI assistants; iii) trust in AI-generated code; iv) perceived productivity improvements; v) reasons for not posting questions publicly; and vi) concerns regarding community sustainability.

Longitudinal statistical analysis: To better understand the temporal relationship between AI adoption and discussion decline, we plan to apply more rigorous longitudinal data analysis. These techniques may help determine whether changes in discussion activity temporally coincide with increases in AI-assisted development indicators.

6 Related Work

Licorish et al. [8] investigated the motivations driving users to interact with question-and-answer (Q&A) platforms before what they describe as the age of LLMs. Through a qualitative analysis, they identified that contributors use these platforms for both advertising and problem solving, with different user profiles ranging from professional usage to learning-oriented purposes. Staegemann et al. [13] investigated, through a literature review, how the rise of ChatGPT (and, consequently, other LLM-based tools) has impacted the sustainability of Stack Overflow. Their findings indicate that the literature already addresses several aspects of this phenomenon. However, the authors also highlight that existing studies remain limited in scope and call for broader investigations, which our study aims to support. Helic and Santos [6] highlighted that engagement on Stack Overflow has declined following the emergence of LLMs. However, they also identified evidence that these platforms continue to be valuable, as contributors appear to be raising the level of discussion by asking more complex questions and contributing to more structured knowledge creation. Taken together, these studies reinforce the relevance of our investigation and corroborate our preliminary findings. To the best of our knowledge, no existing study has explicitly investigated the impact of AI on discussions within the ROS community.

7 Conclusion

This paper presented an initial investigation into the possible relationship between the rise of AI-assisted software development and the decline of public technical discussions within the ROS ecosystem. Although the current results are still preliminary and do not establish causality, they raise important concerns regarding the sustainability of collaborative knowledge sharing in open-source software communities. In particular, the emergence of "silent communities", where developers increasingly rely on private interactions with AI systems instead of public human discussions, may reduce the visibility of real-world development challenges, limit community-driven learning, and affect the long-term availability of technical knowledge for both practitioners and researchers. We also provide detailed plan for future work to better understand the data we collected.

Artifact Availability

We make all the data and analysis artifacts publicly available on GitHub as a replication package[7].

Acknowledgment

We exclusively used ChatGPT to correct part of the grammar, with no content being deliberately generated, and all the text being supervised by human beings.

References

[1] Anton Barua, Stephen W Thomas, and Ahmed E Hassan. 2014. What are developers talking about? An analysis of topics and trends in Stack Overflow. *Empirical Software Engineering* 19, 3 (2014), 619–654. doi:10.1007/s10664-012-9231-y

[2] Emily M. Bender and Alex Hanna. 2025. *The AI Con: How to Fight Big Tech's Hype and Create the Future We Want.* Harper, New York. https://thecon.ai/

[3] Ghizlane Boukili, Said EL Garouani, and Jamal Riffi. 2026. A dataset for human-written and AI-generated code source classification. *Data in Brief* (2026), 112527. doi:10.1016/j.dib.2026.112527

[4] Sunhao Dai, Chen Xu, Shicheng Xu, Liang Pang, Zhenhua Dong, and Jun Xu. 2024. Bias and Unfairness in Information Retrieval Systems: New Challenges in the LLM Era. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Barcelona, Spain) *(KDD '24)*. Association for Computing Machinery, New York, NY, USA, 6437–6447. doi:10.1145/3637528.3671458

[5] Tyna Eloundou, Sam Manning, Pamela Mishkin, and Daniel Rock. 2024. GPTs are GPTs: Labor market impact potential of LLMs. *Science* 384, 6702 (2024), 1306–1308. doi:10.1126/science.adj0998

[6] Denis Helic and Tiago Santos. 2025. Stack Overflow Is Not Dead Yet: Crowd Answers Still Matter. arXiv:2509.05879 [cs.CY] doi:10.48550/arXiv.2509.05879

[7] Junfeng Jiao, Saleh Afroogh, Yiming Xu, and Connor Phillips. 2025. Navigating LLM ethics: advancements, challenges, and future directions. *AI and Ethics* 5, 6 (01 Dec 2025), 5795–5819. doi:10.1007/s43681-025-00814-5

[8] Sherlock A. Licorish, Elijah Zolduoarrati, Tony Savarimuthu, Rashina Hoda, Ronnie De Souza Santos, and Pankajeshwara Sharma. 2026. Why Do You Contribute to Stack Overflow? Understanding Cross-Cultural Motivations and Usage Patterns before the Age of LLMs. arXiv:2603.05043 [cs.SE] doi:10.48550/arXiv.2603.05043

[9] Xuliang Liu and Hao Zhong. 2018. Mining stackoverflow for program repair. In *2018 IEEE 25th International Conference on Software Analysis, Evolution and Reengineering (SANER)*. 118–129. doi:10.1109/SANER.2018.8330202

[10] Ivano Malavolta, Grace A. Lewis, Bradley Schmerl, Patricia Lago, and David Garlan. 2021. Mining guidelines for architecting robotics software. *Journal of Systems and Software* 178 (2021), 110969. doi:10.1016/j.jss.2021.110969

[11] Serhii Nazarovets and Jaime A. Teixeira da Silva. 2025. ChatGPT as an "author": Bibliometric analysis to assess the validity of authorship. *Accountability in Research* 32, 7 (2025), 1148–1158. doi:10.1080/08989621.2024.2345713

[7] https://github.com/IntelAgir-Research-Group/ROS-Index

[12] Marc Pichler, Bernhard Dieber, and Martin Pinzger. 2019. Can I depend on you? Mapping the dependency and quality landscape of ROS packages. In *2019 third IEEE international conference on robotic computing (IRC)*. IEEE, 78–85.

[13] Daniel Staegemann, Mariia Rizun, Christian Haertel, Matthias Pohl, Christian Daase, and Klaus Turowski. 2025. How did the Emergence of ChatGPT Impact Stack Overflow?–A Literature Review. In *Empowering the Interdisciplinary Role of ISD in Addressing Contemporary Issues in Digital Transformation: How Data Science and Generative AI Contributes to ISD (ISD2025 Proceedings)*. University of Gdańsk, Department of Business Informatics & University of Belgrade, Faculty of Organizational Sciences, Belgrade, Serbia. doi:10.62036/ISD.2025.25

[14] Christoph Treude and Martin P. Robillard. 2016. Augmenting API documentation with insights from stack overflow. In *Proceedings of the 38th International Conference on Software Engineering* (Austin, Texas) *(ICSE '16)*. Association for Computing Machinery, New York, NY, USA, 392–403. doi:10.1145/2884781.2884800

[15] Yisheng Zhong, Yizhu Wen, Junfeng Guo, Mehran Kafai, Heng Huang, Hanqing Guo, and Zhuangdi Zhu. 2025. Web Intellectual Property at Risk: Preventing Unauthorized Real-Time Retrieval by Large Language Models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (Eds.). Association for Computational Linguistics, Suzhou, China, 17211–17224. doi:10.18653/v1/2025.emnlp-main.870