Incivility in Open Source Projects: A Comprehensive Annotated Dataset of Locked GitHub Issue Threads

Ramtin Ehsani
Drexel University
Philadelphia, PA, USA
ramtin.ehsani@drexel.edu

Mia Mohammad Imran
Virginia Commonwealth University
Richmond, VA, USA
imranm3@vcu.edu

Robert Zita
Elmhurst University
Elmhurst, IL, USA
rzita8729@365.elmhurst.edu

Kostadin Damevski
Virginia Commonwealth University
Richmond, VA, USA
kdamevski@vcu.edu

Preetha Chatterjee
Drexel University
Philadelphia, PA, USA
preetha.chatterjee@drexel.edu

ABSTRACT

In the dynamic landscape of open source software (OSS) development, understanding and addressing incivility within issue discussions is crucial for fostering healthy and productive collaborations. This paper presents a curated dataset of 404 locked GitHub issue discussion threads and 5961 individual comments, collected from 213 OSS projects. We annotated the comments with various categories of incivility using Tone Bearing Discussion Features (TBDFs), and, for each issue thread, we annotated the triggers, targets, and consequences of incivility. We observed that *Bitter frustration*, *Impatience*, and *Mocking* are the most prevalent TBDFs exhibited in our dataset. The most common triggers, targets, and consequences of incivility include *Failed use of tool/code or error messages*, *People*, and *Discontinued further discussion*, respectively. This dataset can serve as a valuable resource for analyzing incivility in OSS and improving automated tools to detect and mitigate such behavior.

CCS CONCEPTS

• **Software and its engineering** → *Software organization and properties.*

KEYWORDS

OSS, GitHub, developer conversations, incivility

ACM Reference Format:
Ramtin Ehsani, Mia Mohammad Imran, Robert Zita, Kostadin Damevski, and Preetha Chatterjee. 2024. Incivility in Open Source Projects: A Comprehensive Annotated Dataset of Locked GitHub Issue Threads. In *21st International Conference on Mining Software Repositories (MSR '24), April 15–16, 2024, Lisbon, Portugal.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3643991.3644887

1 INTRODUCTION

Issue trackers are pivotal in open source software (OSS) projects, facilitating effective monitoring, organization, and management of work [2]. They serve as a central hub for diverse user and developer feedback, spanning ideas, tasks, features, and bug reports [8]. Within issue discussion threads, interactions range from constructive exchanges to unhealthy, uncivil, and toxic behaviors, which can hinder developer participation and productivity [21, 25, 33]. Existing research has consistently highlighted the adverse impacts of toxicity and incivility within collaborative spaces, with consequences ranging from project abandonment to lower contribution rates [19, 33, 41], especially impacting people from underrepresented communities [5, 34, 43]. Furthermore, these kinds of negative interactions can even affect developers' mental health, resulting in conditions such as stress and burnout [37].

Understanding negative interactions in software projects has gained significant attention in recent years [20, 23, 37, 41]. For instance, Raman et al. developed an automated tool to detect toxicity in GitHub issue threads [37]. Building on Raman et al.'s work, Sarker et al. created a tool for categorizing GitHub and Gitter code review messages as toxic or non-toxic [41]. Complementing these automated approaches, Miller et al. conducted a qualitative analysis of 100 GitHub locked issue threads [33]. In contrast to established categories of toxicity in other domains [7, 10, 12, 26, 31, 38], they identified nuanced and distinct causes of toxicity in OSS, such as *entitlement* and *arrogance*. In exploring the broader category of incivility, Ferreira et al. analyzed Linux mailing lists and GitHub issue threads [21, 23] to study Tone Bearing Discussion Features (TBDFs) – conversational characteristics demonstrated in a written sentence that convey a mood or style of expression.

In spite of the prior work in this area, there is still a lack of a robust and comprehensive approach to address uncivil interactions in OSS. Three key factors strongly contribute to this deficiency: (a) the scarcity of large annotated software engineering-specific datasets [24, 29], (b) a lack of deep understanding of the nuances of negative behavior in OSS (e.g., triggers, targets, and consequences) [33], and (c) a lack of a comprehensive taxonomy consolidating different types of uncivil behaviors in SE. In this paper, we annotate and publish an incivility dataset of 404 locked GitHub issue threads (5,961 issue comments) in open-source repositories. To curate the dataset, we gathered issue threads from 213 projects on GitHub, which had at least 50 contributors. We gathered issues that were either explicitly labeled and locked as "too heated" or demonstrated clear characteristics indicative of heated discussions. After collecting this initial dataset, we manually analyzed

MSR '24, April 15–16, 2024, Lisbon, Portugal
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0587-8/24/04.
https://doi.org/10.1145/3643991.3644887

and annotated the following attributes of incivility in open source: types of incivility, its triggers, associated targets, and consequences or aftermath of incivility as witnessed within these conversations. To support the high-quality annotation of these various attributes related to incivility, we developed an annotation tool using Streamlit [4], a Python library, to streamline the process for the annotators involved in this study.

Our annotated dataset reveals prevalent forms of incivility within OSS projects, with *Bitter frustration*, *Impatience*, and *Mocking* emerging as the most recurrent types. Additionally, the most frequent triggers, targets, and consequences are *Failed use of code*, *People*, and *Discontinued further discussion*, respectively. This nuanced understanding, derived from our dataset, provides a valuable foundation for future research to delve deeper into the intricacies of incivility within OSS communications. Analyzing its unique nature can pave the way for the development of targeted mitigation and detection tools, fostering more inclusive and collaborative environments within these communities. Our dataset is available on GitHub: https://github.com/vcu-swim-lab/incivility-dataset.

2 METHODOLOGY

In this section, we detail our approach for curating an OSS incivility dataset. Our decision to focus on incivility rather than toxicity is driven by a deliberate choice. While these two concepts overlap, toxicity primarily involves language that harms others. Incivility, on the other hand, has a broader scope, encompassing issues that can disrupt constructive and technical discussions [23]. As highlighted by Sadeque et al. [39], the development of a fine-grained incivility detection tool presents a more intricate challenge compared to toxicity detection. We aim to provide a more nuanced understanding of negative discourse dynamics within the context of OSS projects.

2.1 Data Collection

In an effort to mitigate unproductive discussions, GitHub offers functionality that enables project maintainers to lock issue threads, thus preventing further discussion. During the locking process, maintainers can, but are not required to, label the reason the discussion was locked, e.g., as “too-heated”. These moderation tools serve the dual purpose of aiding project contributors in effectively managing and moderating discussions, while also intervening to stop overly contentious conversations when necessary. Locked and labeled discussions within OSS projects offer valuable research data. Since these labels are typically assigned by project maintainers, they serve as a reliable means to pinpoint specific conversation instances of interest to researchers.

Given our goal of curating GitHub issue threads likely to exhibit incivility, particularly within the context of active OSS projects, we established the following data collection procedure:

- The GitHub project must have a minimum of 50 contributors.
- The GitHub issues must have been created in the last 10 years, i.e., between “2013-04-07” and “2023-10-24”.
- The issues must be locked and labeled as “too-heated”, “off-topic” or “spam”. In cases where the issue was locked but the reason (label) was not explicitly stated, we selected the issues that included the text “code of conduct” or “marked as abusive” within the discussion. Such terms have previously been used as clear indicators of potentially toxic or uncivil discussions [33].

Since it was not possible to directly retrieve all the locked issues with the chosen labels from GitHub, we adopted two approaches to ensure the collection of as many of these issues as possible:

GitHub API. Leveraging GitHub's official API [3], we accessed publicly available repositories one-by-one to examine their issue threads. This process was constrained by the rate-limiting mechanism of the API, and thus, given a limited time budget, we were only able to collect data from approximately 600k OSS projects.

GitHub Archive. We used the GH Archive [1], a project dedicated to the recording of the public GitHub timeline and making it readily available for rapid querying and analysis. We used the BigQuery interface and the latest archive (up to the end of 2022) to retrieve GitHub issues where the locked issue type is labeled as “too-heated", “spam”, or “off-topic”.

2.2 Data Selection

Locked issue threads labeled as “too-heated” are the most evident candidates for inclusion in our dataset, due to the prevalence of uncivil conversations within them. Since it is possible for project maintainers to mislabel issues [22], the first two authors examined the collected issues labeled as “too-heated” and removed the ones that were obviously mislabeled.

Our motivation to extend our selection to “spam” and “off-topic” locked issue threads arises from the limited number of “too-heated” threads on GitHub. To select issue threads labeled as “off-topic” or “spam” that have potentially uncivil content, the first two authors examined 422 issue threads from these two categories, manually assessing their content. This effort allowed us to filter out issue threads that were solely “spam” or “off-topic” but not uncivil.

In total, following the application of our selection criteria to the collected repositories and issue threads, merging duplicate instances between the results of the two approaches, and manually filtering the noisy conversations in our dataset, we successfully gathered 404 instances – 338 labeled as “too-heated”, 21 occurrences of “spam”, and 33 labeled as “off-topic”, and 12 instances of issue threads locked without specified reasons but containing the keywords “code of conduct” or “marked as abusive”.

2.3 Annotation Tool

To simplify and streamline the annotation process, we designed a bespoke web annotation tool. We used Streamlit, an open-source app framework, that enables the creation of web apps using data scripts. Our annotation app includes a secure login page, allowing annotators to access the tool with their unique login ID. To aid annotators, we incorporated annotation instructions into the tool's interface, ensuring ready access to category definitions. The GitHub issues are securely stored in an SQLite database, with a subset distributed to each annotator. The web app screenshots and source code are available in our dataset's repository.

2.4 Incivility Categories

For each issue thread in our dataset, we manually annotated four categories: *type of incivility*, *trigger*, *target*, and *consequence*. Each

Table 1: Uncivil Features; Sources are Color Coded for Ferreira et al. , Sarker et al. , and Miller et al. (N=No. of Issues)

Feature	Definition and Example	Most Common Triggers	Most Common Targets	Most Common Consequences	# of Issue Comments
Bitter frustration	***Def.*** *expressing strong frustration* ***e.g.*** *Fixing clippy warnings isn't adding anything for users*	Failed use of tool/code or error messages (N=49), Technical disagreement (N=49)	People (N=107), Code/tool (N=54)	Discontinued further discussion (N=71), Escalating further (N=63), Provided technical explanation (N=37)	**492**
Impatience	***Def.*** *express a feeling that it is taking too long* ***e.g.*** *I am locking this thread. It is becoming useless*	Technical disagreement (N=34), Failed use of tool/code or error messages (N=29),	People (N=70), Code/tool (N=37)	Discontinued further discussion (N=49), Escalating further (N=36), Provided technical explanation (N=31)	264
Mocking	***Def.*** *making fun of someone else* ***e.g.*** *congrats, you won an award for the best support of the month*	Failed use of tool/code or error messages (N=19), Technical disagreement (N=15), Communication breakdown (N=14)	People (N=59), Code/tool (N=12)	Escalating further (N=37), Discontinued further discussion (N=32) Provided technical explanation (N=20)	180
Irony	***Def.*** *signify the opposite in a mocking way* ***e.g.*** *Ok, you win, have fun arguing forever instead of proposing a solution*	Technical disagreement (N=11), Failed use of tool/code or error messages (N=10)	People (N=20), Code/tool (N=16)	Escalating further (N=21), Discontinued further discussion (N=19), Provided technical explanation (N=12)	64
Vulgarity	***Def.*** *using profanity or improper language* ***e.g.*** *It honestly looks like they don't give a sh*t, rules this out as an option for me!*	Failed use of tool/code or error messages (N=13), Technical disagreement (N=9)	People (N=31), Code/tool (N=12)	Escalating further (N=26), Discontinued further discussion (N=16), Trying to stop the incivility (N=12)	71
Threat	***Def.*** *put a condition impacting the result of discussion* ***e.g.*** *This is the final notice. Be honest, respectable, and collaborative*	Communication breakdown (N=4), Violation of community conventions (N=3)	People (N=10), Code/tool (N=5)	Discontinued further discussion (N=10), Escalating further (N=8), Trying to stop the incivility (N=7)	23
Entitlement	***Def.*** *expecting special privileges* ***e.g.*** *Or you could start contributing instead of bashing people who actually do the work*	Technical disagreement (N=12), Failed use of tool/code or error messages (N=9)	People (N=30), Code/tool (N=10)	Escalating further (N=20), Discontinued further discussion (N=20), Provided technical explanation (N=12)	69
Insulting	***Def.*** *remarks directed at another person* ***e.g.*** *Seems like only thing you can do so far is talk, come back when you will have any skill to show.*	Failed use of tool/code or error messages (N=23), Technical disagreement (N=19)	People (N=57), Code/tool (N=19)	Escalating further (N=41), Discontinued further discussion (N=33), Provided technical explanation (N=17)	174
Identity attacks/ Name-calling	***Def.*** *Race, Religion, Nationality, Gender, Sexual-oriented attacks* ***e.g.*** *I would not be surprised if this database is maintained by the Russians*	Politics/ideology (N=6) Failed use of tool/code or error messages (N=5)	People (N=15), Company/organization (N=7)	Escalating further (N=13), Discontinued further discussion (N=8), Invoke Code of Conduct (N=5)	28

category was selected based on existing literature on harmful interactions across various domains such as social media [6, 15, 36, 44], online gaming communities [30, 32], and software engineering [13, 14, 17, 19, 21, 23, 25, 33, 41, 42]. Using a combination of deductive and inductive coding methods, we refined the feature set for each category to enhance the quality of our annotations [9, 11]. This involved an iterative process, merging similar features and eliminating those that were too general or irrelevant, until we achieved a comprehensive set of features for each annotation category.

Types of Incivility. To annotate types of incivility in our dataset, we utilized Ferreira et al.'s framework [23] for identifying uncivil textual elements, known as Tone Bearing Discussion Features (TBDFs), initially introduced by Coe et al [16]. TBDFs include conversational attributes in written sentences that convey specific moods or expressive styles. According to Ferreira et al. [23], these characteristics are categorized into four sets: positive, negative, neutral, and uncivil. We focused solely on the uncivil characteristics for annotation. We also incorporated elements from the works of Miller et al. [33] and Sarker et al. [41] to further enhance our feature set's comprehensiveness. The complete set of annotation categories in our dataset, along with their definitions and examples, is listed in Table 1.

Trigger. Our aim was to identify the underlying triggers of incivility within GitHub conversations. The categorizations for this annotation were drawn from the extensive research of Ferreira et al. [23], which includes triggers like *Communication breakdown*, *Rejection*, and *Violation of community conventions*, and Miller et al. [33], which includes triggers such as *Failed use of tool/code or error messages*, *Past interactions*, *Politics/ideology*, and *Technical disagreement.* We expanded these categories by introducing an additional category, *Unprovoked*, to capture instances of incivility without a discernible trigger.

Target. This annotation aimed to pinpoint the specific target of incivility in conversations. The categorizations were derived from Miller et al. [33], including *People*, *Code/tool*, *Company/organization*, *Self-directed*, and *Undirected.*

Consequence. The primary goal of this annotation was to uncover the repercussions of incivility as observed within conversations, thereby shedding light on the aftermath of such interactions in developer communications. These categorizations were inspired by existing research by Ferreira et al. [23], which includes consequences such as *Discontinued further discussion*, *Provided technical explanation*, *Accepting criticism*, and *Trying to stop the incivility*, and Miller et al. [33], which includes *Invoke Code of Conduct*, *Turning constructive*, and *Escalating further.*

2.5 Data Annotation Procedure

A total of 19 university students (junior undergraduate=2, senior undergraduate=16, master's=1) studying computer science were recruited as annotators. These students were recruited through email outreach. After completing a consent form, clear annotation instructions, including examples, were provided in the form of an external document (with key parts of it repeated in the annotation tool as reminders). All annotators had prior experience with GitHub (0-2 years=13, 2-4 years=5, 4+ years=1), and one annotator had prior experience contributing to open source projects. Each student annotated approximately 20 issue threads.

To further improve the annotation's quality, we use GPT-4 [35] (via the 'gpt-4' API), which has shown promising results in text annotation, in some cases outperforming humans [18, 27, 28]. We formulate a prompt in which GPT-4 systematically evaluates student annotations utilizing a 5-point scale. Instances with an agreement score below 3 are considered disagreements in our analysis. We found 549 out of 5,961 utterances where GPT-4 disagreed with the human annotation. Two of the authors of this paper, manually checked those 549 utterances to resolve the disagreements, revising the annotation of 311 instances. For example, the comment *"Calm down, Please."* was initially labeled as *Impatience.* However, upon a manual review and considering the context of the ongoing conversation within the issue thread, the annotation was revised to *None.* This process ensured the high accuracy and reliability of the final annotated dataset.

3 DATASET DESCRIPTION

Incivility Annotations. Of the 5,961 issue comments analyzed, 1,365 were annotated with an incivility feature. The distribution of these annotations is detailed in Table 1. *Bitter frustration*, *Impatience*, and *Mocking* are the most recurrent uncivil features in this dataset. Among the 404 issue threads, 319 have at least one uncivil feature annotation. Of the 85 threads without any identified uncivil feature, 78 were locked as "too heated," 2 as "spam," 2 without specified reasoning, and 1 as "off-topic."

Issue Thread Annotations. The distribution of annotated triggers, targets, and consequences within this dataset is presented in Figure 1. *Failed use of tool/code or error messages*, *Technical disagreement*, and *Communication breakdown* are the most prevalent triggers. The most frequent targets are *People* and *Code/tool*, while the most common consequences include *Discontinued further discussion*, *Escalating further*, and *Provided technical explanation.* Notably, none of the uncivil issue threads in this dataset transitioned into constructive discussions (*Turning Constructive N=0*). Annotators could label each issue with multiple consequences, with combinations like *[Escalating further, Discontinued further discussion]*, *[Invoke Code of Conduct, Discontinued further discussion]*, and *[Escalating further, Trying to stop the incivility]* being prominent.

Observations. When uncivil discussions target people, the main triggers are *Communication breakdown (N=33)*, *Technical disagreement (N=27)*, and *Failed use of tool/code or error messages (N=21)*, with the most common consequences being *Discontinued further discussion (N=62)*, *Escalating further (N=51)*, and *Trying to stop the incivility (N=27).* In contrast, when incivility targets Code/tool, the primary triggers are *Failed use of tool/code or error messages (N=32)*, *Technical disagreement (N=22)*, and *Communication breakdown (N=6)*, with the most frequent consequences being *Discontinued further discussion (N=34)*, *Provided technical explanation (N=15)*, and *Escalating further (N=14).* An interesting finding is that *Failed use of tool/code or error messages* as a trigger often leads to incivility directed at *Code/tool*, whereas *Technical disagreement* usually results in incivility aimed at *People.* Figure 1 illustrates detailed relationships between targets, triggers, and consequences in this dataset, such as *Communication breakdown* typically targeting *People* and leading to the discontinuation of further discussion.

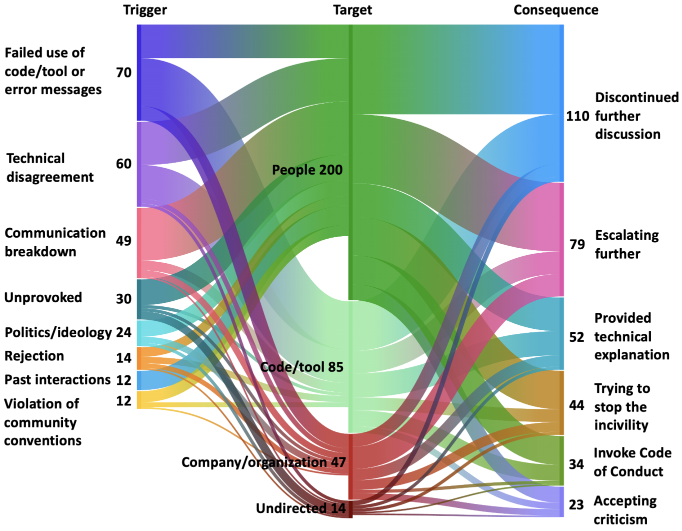

Figure 1: Triggers, Targets, and Consequences of Incivility

4 RESEARCH OPPORTUNITIES

Our dataset presents numerous opportunities for addressing and exploring challenges in sustainable software projects and developer productivity. The prevalence of toxic interactions and uncivil language within OSS communities has become a pressing issue, leading to negative emotional experiences and developer isolation. This dataset is a valuable resource for conducting comprehensive analyses of incivility within developer communications.

It offers the potential to train and refine automatic incivility detection tools. These tools can identify uncivil conversations and help mitigate disruptive interactions within discussions. Our annotations provide more than just flags for uncivil comments; they offer insights into the specific types of incivility present. This can be used to analyze developer interactions, highlighting the prevalence and nuances of different incivility types within developer communications. Previous research indicates that tools trained in other domains, like the Google Perspective API, are ineffective for software engineering (SE) corpora due to the unique nature of SE text [37, 40]. Thus, developing an SE-specific incivility detection tool that understands the nuances of developer conversations, including SE jargon, is a valuable contribution to both the literature and the OSS community.

We focused our analysis on popular OSS projects on GitHub with significant contributor numbers. This approach allows us to examine the dynamics of incivility within these projects, identifying primary factors contributing to such occurrences, especially analyzing triggers, targets, and consequences of uncivil conversations in OSS.

Furthermore, our dataset enables exploration of how incivility might impact key project attributes and overall project health, including code quality and commit frequency. By employing triangulation studies or integrating data from GitHub's version control, we can assess the effects of incivility on developers' code quality and commits, providing a deeper understanding of team dynamics.

Code of Conduct is often used in OSS moderation. The dataset could also further enable analysis of moderation strategies and policies adopted by different open source projects to handle incivility.

This dataset may help to forecast when a conversation is going to derail. The triggers and targets annotated in the dataset provide a foundation to explore personalized intervention approaches when automated tools detect potential early signs of uncivil conversations arising.

Additionally, considering the ongoing challenge of underrepresentation in OSS development, our dataset offers a unique opportunity to investigate how incivility affects individuals from underrepresented communities. By incorporating considerations of gender, race, and cultural aspects, and given the substantial populations of these projects, we can explore the implications of incivility on these communities.

REFERENCES

[1] 2023. *GitHub Archive.* https://www.gharchive.org/

[2] 2023. *GitHub Issues.* https://github.com/features/issues

[3] 2023. *GitHub Repositories.* https://api.github.com/repositories

[4] 2023. *Streamlit.* https://streamlit.io/

[5] Khaled Albusays, Pernille Bjorn, Laura Dabbish, Denae Ford, Emerson Murphy-Hill, Alexander Serebrenik, and Margaret-Anne Storey. 2021. The Diversity Crisis in Software Development. *IEEE Software* 38, 2 (2021), 19–25.

[6] Ashwaq Alsoubai, Jihye Song, Afsaneh Razi, Nurun Naher, Munmun De Choudhury, and Pamela J. Wisniewski. 2022. From 'Friends with Benefits' to 'Sextortion:' A Nuanced Investigation of Adolescents' Online Sexual Risk Experiences. 6, CSCW2, Article 411 (nov 2022), 32 pages. https://doi.org/10.1145/3555136

[7] Amna Anjum, Xu Ming, Ahmed Faisal Siddiqi, and Samma Faiz Rasool. 2018. An Empirical Study Analyzing Job Productivity in Toxic Workplace Environments. *International Journal of Environmental Research and Public Health* 15, 5 (May 2018), 1035. https://doi.org/10.3390/ijerph15051035

[8] Deeksha M. Arya, Wenting Wang, Jin L. C. Guo, and Jinghui Cheng. 2019. Analysis and Detection of Information Types of Open Source Software Issue Discussions. *CoRR* abs/1902.07093 (2019). arXiv:1902.07093 http://arxiv.org/abs/1902.07093

[9] Theophilus Azungah and Rulinawaty Kasmad. 2020. Qualitative Research Journal Article information: For Authors Qualitative research: deductive and inductive approaches to data analysis. (08 2020).

[10] Gary Blau and Lynne Andersson. 2005. Testing a measure of instigated workplace incivility. *Journal of Occupational and Organizational Psychology* 78, 4 (2005), 595–614.

[11] Elizabeth H. Bradley, Leslie A. Curry, and Kelly J. Devers. 2007. Qualitative data analysis for health services research: developing taxonomy, themes, and theory. *Health Services Research* 42, 4 (Aug. 2007). https://doi.org/10.1111/j.1475-6773.2006.00684.x

[12] Vikas S Chavan and Shylaja S S. 2015. Machine learning approach for detection of cyber-aggressive comments by peers on social media network. In *2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI).* 2354–2358. https://doi.org/10.1109/ICACCI.2015.7275970

[13] Jithin Cheriyan, Bastin Tony Roy Savarimuthu, and Stephen Cranefield. 2020. Norm violation in online communities - A study of Stack Overflow comments. *CoRR* abs/2004.05589 (2020). arXiv:2004.05589 https://arxiv.org/abs/2004.05589

[14] Jithin Cheriyan, Bastin Tony Roy Savarimuthu, and Stephen Cranefield. 2021. Towards Offensive Language Detection and Reduction in Four Software Engineering Communities. In *Proceedings of the 25th International Conference on Evaluation and Assessment in Software Engineering* (Trondheim, Norway) *(EASE '21).* Association for Computing Machinery, New York, NY, USA, 254–259.

[15] Usman Chohan and Aron D'Souza. 2020. A Critical Appraisal of the Twitterverse. *SSRN Electronic Journal* (01 2020). https://doi.org/10.2139/ssrn.3546890

[16] Kevin Coe, Kate Kenski, and Stephen Rains. 2014. Online and Uncivil? Patterns and Determinants of Incivility in Newspaper Website Comments. *Journal of Communication* 64 (08 2014). https://doi.org/10.1111/jcom.12104

[17] Sophie Cohen. 2021-08-18. Contextualizing toxicity in open source: a qualitative study. In *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering* (New York, NY, USA) *(ESEC/FSE 2021).* Association for Computing Machinery, 1669–1671. https://doi.org/10.1145/3468264.3473492

[18] Haixing Dai, Zhengliang Liu, Wenxiong Liao, Xiaoke Huang, Zihao Wu, Lin Zhao, Wei Liu, Ninghao Liu, et al. 2023. Chataug: Leveraging chatbot for text data augmentation. *arXiv preprint arXiv:2302.13007* (2023).

[19] Carolyn D Egelman, Emerson Murphy-Hill, Elizabeth Kammer, Margaret Morrow Hodges, Collin Green, Ciera Jaspan, and James Lin. 2020. Predicting developers' negative feelings about code review. In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering.* 174–185.

[20] Ramtin Ehsani, Rezvaneh Rezapour, and Preetha Chatterjee. 2023. Exploring Moral Principles Exhibited in OSS: A Case Study on GitHub Heated Issues. In *Proceedings of the 31st ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering.* 2092–2096.

[21] Isabella Ferreira, Bram Adams, and Jinghui Cheng. 2022. How heated is it?: Understanding GitHub locked issues. In *Proceedings of the 19th International Conference on Mining Software Repositories.* ACM. https://doi.org/10.1145/3524842.3527957

[22] Isabella Ferreira, Bram Adams, and Jinghui Cheng. 2022-05-23. How heated is it? Understanding GitHub locked issues. In *Proceedings of the 19th International Conference on Mining Software Repositories.* 309–320. https://doi.org/10.1145/3524842.3527957 arXiv:2204.00155 [cs]

[23] Isabella Ferreira, Jinghui Cheng, and Bram Adams. 2021-10-13. The "Shut the f**k up" Phenomenon: Characterizing Incivility in Open Source Code Review Discussions. 5 (2021-10-13), 1–35. Issue CSCW2. https://doi.org/10.1145/3479497 arXiv:2108.09905 [cs]

[24] Isabella Ferreira, Ahlaam Rafiq, and Jinghui Cheng. 2022-07-07. Incivility Detection in Open Source Code Review and Issue Discussions. https://doi.org/10.2139/ssrn.4156317

[25] Daviti Gachechiladze, Filippo Lanubile, Nicole Novielli, and Alexander Serebrenik. 2017. Anger and Its Direction in Collaborative Software Development. In *2017 IEEE/ACM 39th ICSE-NIER.* 11–14. https://doi.org/10.1109/ICSE-NIER.2017.18

[26] Rocío Galarza Molina and Freddie Jennings. 2017. The Role of Civility and Metacommunication in Facebook Discussions. *Communication Studies* (11 2017), 1–25. https://doi.org/10.1080/10510974.2017.1397038

[27] Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. 2023. Chatgpt outperforms crowd-workers for text-annotation tasks. *arXiv preprint arXiv:2303.15056* (2023).

[28] Fan Huang, Haewoon Kwak, and Jisun An. 2023. Is ChatGPT better than Human Annotators? Potential and Limitations of ChatGPT in Explaining Implicit Hate Speech. In *Companion Proceedings of the ACM Web Conference 2023.*

[29] Mia Mohammad Imran, Yashasvi Jain, Preetha Chatterjee, and Kostadin Damevski. 2022. Data Augmentation for Improving Emotion Recognition in Software Engineering Communication. In *37th IEEE/ACM International Conference on Automated Software Engineering.*

[30] Yubo Kou. 2020-11-03. Toxic Behaviors in Team-Based Competitive Gaming: The Case of League of Legends. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play* (New York, NY, USA) *(CHI PLAY '20).* Association for Computing Machinery, 81–92. https://doi.org/10.1145/3410404.3414243

[31] Haewoon Kwak, Jeremy Blackburn, and Seungyeop Han. 2015. Exploring Cyberbullying and Other Toxic Behavior in Team Competition Online Games. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15).* Association for Computing Machinery, New York, NY, USA, 3739–3748. https://doi.org/10.1145/2702123.2702529

[32] Haewoon Kwak, Jeremy Blackburn, and Seungyeop Han. 2015. Exploring Cyberbullying and Other Toxic Behavior in Team Competition Online Games. 22 (04 2015). https://doi.org/10.1145/2702123.2702529

[33] Courtney Miller, Sophie Cohen, Daniel Klug, Bogdan Vasilescu, and Christian Kästner. 2022. "Did You Miss My Comment or What?" Understanding Toxicity in Open Source Discussions. In *2022 IEEE/ACM 44th International Conference on Software Engineering (ICSE).* 710–722. https://doi.org/10.1145/3510003.3510111

[34] Dawn Nafus. 2012. 'Patches don't have gender': What is not open in open source software. *New Media & Society* 14, 4 (2012), 669–683. https://doi.org/10.1177/1461444811422887 arXiv:https://doi.org/10.1177/1461444811422887

[35] OpenAI. 2023. GPT-4 Technical Report. arXiv:2303.08774 [cs.CL]

[36] Juergen Pfeffer, T. Zorbach, and Kathleen Carley. 2013. Understanding online firestorms: Negative word-of-mouth dynamics in social media networks. *Journal of Marketing Communications* 20 (12 2013), 117–128. https://doi.org/10.1080/13527266.2013.797778

[37] Naveen Raman, Minxuan Cao, Yulia Tsvetkov, Christian Kästner, and Bogdan Vasilescu. 2020. Stress and Burnout in Open Source: Toward Finding, Understanding, and Mitigating Unhealthy Interactions. In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: New Ideas and Emerging Results* (Seoul, South Korea) *(ICSE-NIER '20).* Association for Computing Machinery, New York, NY, USA, 57–60. https://doi.org/10.1145/3377816.3381732

[38] Samma Faiz Rasool, Mansi Wang, Minze Tang, Amir Saeed, and Javed Iqbal. 2021. How Toxic Workplace Environment Effects the Employee Engagement: The Mediating Role of Organizational Support and Employee Wellbeing. *International Journal of Environmental Research and Public Health* 18, 5 (March 2021), 2294. https://doi.org/10.3390/ijerph18052294

[39] Farig Sadeque, Stephen Rains, Yotam Shmargad, Kate Kenski, Kevin Coe, and Steven Bethard. 2019. Incivility detection in online comments. In *Proceedings of the eighth joint conference on lexical and computational semantics (* SEM 2019).*

[40] Jaydeb Sarker, Asif Kamal Turzo, and Amiangshu Bosu. 2020. A Benchmark Study of the Contemporary Toxicity Detectors on Software Engineering Interactions. *2020 27th Asia-Pacific Software Engineering Conference (APSEC)* (2020), 218–227.

[41] Jaydeb Sarker, Asif Kamal Turzo, Ming Dong, and Amiangshu Bosu. 2023. Automated Identification of Toxic Code Reviews Using ToxiCR. *ACM Trans. Softw. Eng. Methodol.* 32, 5, Article 118 (jul 2023), 32 pages. https://doi.org/10.1145/3583562

[42] Bastin Tony Roy Savarimuthu, Zoofishan Zareen, Jithin Cheriyan, Muhammad Yasir, and Matthias Galster. 2023. Barriers for Social Inclusion in Online Software Engineering Communities - A Study of Offensive Language Use in Gitter Projects. In *Proceedings of the 27th International Conference on Evaluation and Assessment in Software Engineering (EASE '23).* ACM, 217–222.

[43] Bianca Trinkenreich, Igor Wiese, Anita Sarma, Marco Gerosa, and Igor Steinmacher. 2022. Women's Participation in Open Source Software: A Survey of the Literature. 31, 4, Article 81 (aug 2022), 37 pages. https://doi.org/10.1145/3510460

[44] Cynthia Van Hee, Gilles Jacobs, Chris Emmery, Bart Desmet, Els Lefever, Ben Verhoeven, Guy Pauw, Walter Daelemans, and Véronique Hoste. 2018. Automatic Detection of Cyberbullying in Social Media Text. (01 2018).