Seeking New Measures for Gender Bias Effects in Open-Source Software

Huilian Sophie Qiu
Carnegie Mellon University
Pittsburgh, PA, USA
hsqq@cmu.edu

Moira Connell
The Ellis School
Pittsburgh, PA, USA
connellm@theellisschool.org

ABSTRACT

The problem of low gender diversity in open-source software (OSS) has been reported and studied in recent years. However, prior studies found that gender bias theories in social sciences cannot help us effectively identify gender bias effects in OSS. Our study takes the first step toward finding new measures for gender bias in OSS. This paper attempts to employ linguistic theories to identify different collaboration patterns between different genders. Our contributions are two-fold: we review linguistic literature on diversity and online collaboration, then we apply linguistic theories from our literature reviews to a random sample of code review conversations on GitHub.

1 INTRODUCTION

The low gender diversity in the open-source software (OSS) community is a well-known phenomenon: among the GitHub users whose genders can be inferred, less than 10% are women [1, 6, 15, 30]. The low gender diversity is problematic as it can threaten OSS sustainability as a whole. Firstly, low gender diversity is suboptimal for project success: studies found that higher gender diversity is associated with fewer community smells [7, 38] and higher team performance [26, 34, 40]. Moreover, the highly imbalanced gender representation and the unwelcoming culture in some open-source projects [23] may discourage underrepresented groups from initial participation, which limits opportunities both for those individuals and for employers that use OSS as a talent pool [32, 33].

One of the reasons for women's low participation is gender bias [19, 23, 39]. Based on interviews with OSS developers, Nafus [23] pointed out that, in OSS, "sexist behavior is [...] as constant as it is extreme." A quantitative study by Terrell et al. [39] reports that female contributors face unfair treatments when making code contributions.

This piece of work builds upon a prior attempt on investigating gender bias effects in OSS by Imtiaz et al. [19]. In their paper, Imtiaz *et al.* adapted a gender bias framework by Williams and Dempsey [42], which was developed for women in the workforce, to the context of OSS. The framework discusses four effects of gender bias women may face in the workforce. *Prove-It-Again*: women must provide more evidence than men to demonstrate their competence. *Tightrope*: women avoid extreme behaviors, *e.g.*, too polite or too impolite, to avoid backlash. *Maternal Wall*: women who are mothers have their commitment and competence questioned. *Tug of War*: women are discouraging to other women.

Using several count-based measures and statistical tests, Imtiaz et al. [19] only found strong evidence supporting Tightrope, where women tend to behave in a more restrained way than men. However, we should not conclude the absence of these bias effects without searching for better or more suitable measures and analyses.

Our vision is to investigate the effects of gender bias deeper by examining the language used in open-source conversations using natural language processing (NLP) techniques. As a sociotechnical activity, OSS development entails many communications in the form of natural language, such as code review discussions. It is, therefore, reasonable to suppose that the use of language in a conversation might provide information about roles, status, and other aspects of an individual or a group's dynamics [9]. For example, Paul et al. [28] analyze the sentiment of the language in code review comments by contributors of different genders and found some significant differences.

We can further investigate that if different language patterns used by different genders can have an impact on the contribution outcome. There is evidence in other domains, such as medical care, where studies show that the use of language can reflect health care providers' implicit bias, which can negatively impact the results of medical care and patient satisfaction, especially among patients of minority groups [18, 43].

We propose a new method to investigate gender bias in open-source communication. We first review linguistics literature to identify potential linguistic features whose usages differ between genders. Using these features, we can run a model to find out what linguistic features are more likely to be associated with a higher success rate in code contribution. If certain female-dominant linguistic features are found to be associated with a lower success rate, then it might suggest the presence of implicit bias.

This paper reports our first step in implementing this method. In Section 2, we discuss linguistic theories that are related to how people of different genders may talk differently in online communications. Because not all linguistic features can be observed in conversations in software engineering, in Section 3, we describe how we sample a small set of pull request (PR) conversations on GitHub and perform qualitative analysis to select relevant linguistic measures. In Section 4, we discuss our findings and in Section 5, we discuss current problems and future plans.

CHASE'22 , May 21–29, 2022, Pittsburgh, PA, USA

ACM ISBN 978-1-4503-9342-3/22/05.
https://doi.org/10.1145/3528579.3529169

2 RELATED WORK

2.1 Gender diversity in open-source software communities

Traditionally, technology is considered a male-dominated field, but recently, OSS communities have started to value diversity of all kinds, including gender. There is empirical evidence to date that suggests that gender diversity is associated with higher team productivity [26, 40] and fewer community smells [7, 38]. Russo and Stol [34] confirmed that mixed-gender software engineering teams are associated with better performance because men and women tend to display different personalities. Moreover, a software team with higher diversity is more likely to understand users' needs better because the developers can more naturally represent a wider group of their intended users [22].

Several studies have identified gender bias in OSS. Terrell et al. [39] showed that women's PRs acceptance rate is lower than their male counterparts if they are outsiders to the project and their gender is visible on their profiles. However, when their gender is not visible, women's acceptance rate is higher than men's. Imtiaz et al. [19] have investigated the gender bias on OSS platforms quantitatively using Williams and Dempsey's framework [42] derived from gender studies literature. They found that the effect of gender bias is largely invisible on GitHub. However, there are still signals of women concentrating their work in fewer places and communicating, compared to men, in a more restrained manner. Wang et al. [41] focused on the competence-conference gap and found that, compared to male developers, female developers are often hesitant to contribute to new projects even when they possess the competence to make valuable contributions.

2.2 Linguistic Politeness Theory

The linguistic politeness theory was developed by Brown and Levinson [3]. Centered on the notion of avoiding face-threatening acts, Brown and Levinson outlined two types of politeness strategies: *positive politeness* and *negative politeness*. A positive politeness approach is oriented towards the hearer's positive face, the want that his or her wants are desirable to some others. Some of the examples include giving complement or showing respect. A negative politeness approach is an attempt to save the hearer's negative face, the want that his or her actions are unimpeded by others, by avoiding placing burdens on the hearers' actions. Some examples include using hedge words and being indirect or apologetic.

The politeness theory has been applied to various research projects to understand communication in online communities. For instance, Danescu-Niculescu-Mizil et al. [10] compiled a list of features, including the use of second-person pronouns and "please", that can be used to classify politeness on platforms such as Wikipedia and Stack Exchange. Burke and Kraut [4] measured and compared the readers' perceived politeness on conversations from 12 online groups on various topics. They found significant differences across groups: in some technical groups, politeness increase reply rates, but in some political groups, rudeness is more effective. Another research by Fangl et al. [13] explored the use of gratitude, one of the politeness strategies, on Stack Overflow and found that gratitude expressions can motivate users to generate content of higher quality. However, few studies examined how gender intersects with politeness strategy in the context of OSS communities.

2.3 Closed-vocabulary methods

A classical approach to analyzing language differences is to rely on a pre-set vocabulary. Mulac et al. [21] provided a literature review on gender differences in language usage and listed words that are found to be more likely to be used by one gender. In their samples, some of the linguistic features more often used by women include negations and hedges. Some of the linguistic feature more often used by men include elliptical sentences and "I" references.

Linguistic Inquiry and Word Count (LIWC) [29] is a widely used list of words for conversation analysis. LIWC contains 2,000 words divided into 74 linguistic categories. Researchers can count the percentage of the number of occurrences of each word in the corpus and use them as features. Using LIWC, Newman et al. [25] coded a large corpus of text and summarized a list of vocabularies with their mean and standard deviation of usage by different genders. For example, they found that negation is more used by women than men and women use more emotion words and third person pronouns.

2.4 Open-vocabulary methods

An open-vocabulary method tries to extract connections or patterns from the data rather than rely on *a priori* vocabulary. Schwartz et al. [36] applied differential language analysis (DLA), an open-vocabulary method, to analyze over 700 million words from a Facebook dataset and compare language use across different identifiers, such as personality, gender, and age. Their analysis found that female users used more emotion-related words and first-person singulars while men used more swear words and object words (*e.g.*, Xbox). The drawback is that the results from one study may not be directly applicable to other context. In this case, results obtained using Facebook data have limited value in software engineering context.

2.5 Sentiment analysis

Sentiment analysis [27] is a popular technique in the software-engineering community for text analytics, such as issue discussions [14] and pull request comments [17]. Some popular software engineering sentiment analysis tools include Senti4SD [5] and SentiCR [2], and SentiSE [20].

Paul et al. [28] used SentiSE [20] to analyze the sentiment of code review comments from six popular open-source projects. They found that women are less likely to express sentiment than men, and male developers write more frequent negative comments and fewer positive encouragements from their female collaborators.

2.6 Emojis

In recent years, emojis have become increasingly ubiquitous in online communication, and software communities are no exception. Nonverbal symbols have been a very important part of our online communication. As Dresner & Herring [11] states, "They are most often characterized as iconic indicators of emotion, conveyed through a communication channel that is parallel to the linguistic one." As a part of our review of literature on computer-mediated communication, we surveyed literature surrounding emojis as a mode of communication and their specific relevance to gender.

Chen *et al.* [8] analyzed 134,419 anonymized Android users with self-reported genders and their 401 million messages over three months. They found statistically significant differences between women and men usage of emojis: women are more likely to use emojis than men and men and women have different preferences in using emojis to express sentiments.

3 METHODS AND DATA

In this preliminary analysis, we first randomly sampled 100,000 PRs with comments from GHTorrent [16] and used the gender inference tool Namsor [24] to infer the PR author's gender. We use Namsor because it is one of the most accurate gender inference tools based on names [35, 37]. While we acknowledge that gender is not limited to binary, we recognize the difficulty and lack of information regarding non-binary genders in open source. Thus, to make gender identification more quantitatively tractable, we choose to simplify gender identification by assuming binary gender *male* and *female* in this study.

Namsor first infers one's cultural origin using the last name, then infers the gender using the first name. Including one's cultural origin can reduce misclassifications of names such as "Andrea", which is a male given name in Italian but a female given name in English or German. Along with the gender classification outcome, Namsor also provides confidence that a user's gender is correctly identified. We only kept the genders of users whose associated probability is higher than 0.80. The 0.80 threshold retained 83.8% of the gender data.

Due to the constraint of time, at this stage, we picked and coded 4 PRs by men contributors and four by female contributors from our 100,000. We encountered several difficulties when finding suitable PRs for our qualitative analysis. Since we want to analyze how people collaborate and interact, we only use PRs with at least four comments by human users and at least two participants. It is even more challenging to find PRs authored by a female contributor that satisfy our requirements.

For each comment, we mark the presence of linguistic features discussed by Newman et al. [25] and Mulac et al. [21], the two studies that used the closed-vocabulary method and provided a list of linguistic features that are more often observed among one gender than the other. We also augmented our qualitative coding on linguistic usage as we coded the conversations. Finally, we compiled a list of linguistic features present in open-source code review conversations.

4 RESULTS

Table 1 shows the list of linguistic features we found in our sample of PRs. Features are divided into columns "Women" and "Men" according to how previous studies found them to be more often associated with one gender than the other [21, 25]. Since we found conflicting evidence from existing studies [21, 25] on the use of 1st person pronouns, we put it under "Other". Those without a reference to existing literature are the ones we observed when coding our samples. We will briefly discuss them later in this section.

From our small sample of 8 PRs, we were not able to observe any trends in which some of the linguistic features were more often associated with one gender than the other. Since our sample contains only merged PRs, we were not able to test if any of these features were associated with higher or lower success rates either. These two tasks are saved for future work.

Although we could not establish a systematic correlation between linguistic patterns and PR success, in this round of coding, we observed several linguistic phenomena that might be unique to the context of software engineering or even OSS.

4.1 More forms of negative politeness

In the classic politeness framework, negative politeness constitutes speech acts that avoid impositions on the hearer. In our sample of PRs, we observed several negative politeness strategies when providing code reviews or discussing the contribution.

One prevalent example is the use of the word *"nitpick"*. We observed several maintainers use the word "nitpick" when picking out small details that should be changed, *e.g.*, start a variable name with a lower case letter. Egelman *et al.* [12] found that nitpicking was considered by many programmers as one of the causes of interpersonal conflicts. However, we think that the phrase "nitpick" serves as a negative politeness strategy here. By explicitly pointing out the comment is "nitpicking", the reviewer is saving the author's negative face, the want that his or her actions are unimpeded by others, by admitting that the requested change is small and would not overshadow the merit of the contribution.

4.2 Formality

We observed a spectrum of formality in code review conversations. In most cases, the conversation is more casual, as if in a chat. We observed the usage of *slang and colloquialisms*, reflecting that code review conversations sometimes can be less formal and even somewhat casual. However, there are also cases where people use comments similar to emails: they start by calling the name of the hearer using *"@"* in each comment.

4.3 Inviting Suggestions

We found many uses of the phrase "feel free" to provide suggestions or to indicate that suggestions "are welcome," primarily by authors. We observed the occurrences of inviting suggestions to be the same among male and female authors in our small sample. This can be an interesting feature to model computationally since it may reflect the power dynamic between authors and reviewers.

4.4 Technical *vs.* personal

We found that more reviewers of women-authored PRs used second-person pronouns than those of men-authored PRs. In an analysis on n-grams that are overly represented in PRs with interpersonal conflicts and those without, Qiu et al. [31] found that second-person pronouns are more prevalent among conversations with interpersonal conflicts. Therefore, this feature is worth further investigation once our dataset is ready.

5 DISCUSSIONS

There are several problems we are still trying to solve.

Table 1: Linguistic features we observed in open-source conversations

Women	Ref.	Men	Ref.	Other	Ref.
hedgewords	[21, 25]	references to quantity	[21, 25]	1st person	[21, 25]
what-question	[21]	elliptical sentences	[21]	"nitpick"	
intensive adverbs	[21]	2nd person	[25]	@	
emoji	[8]			slang/colloquialisms	
negation	[21, 25]			inviting suggestion	
3rd person	[21, 25]				
references to emotions	[21, 25]				
uncertainty verbs	[21]				
oppositions	[21]				

5.1 Frequency of a linguistic pattern

Considering the frequency of a linguistic pattern can distinguish patterns that are more indicative and representative from the ones that are used only once or twice. Yet, if we consider the frequency, how should we define it? Should it be the number of occurrences of certain linguistic patterns divided by the number of words that person commented? We are still looking for an answer.

5.2 Dataset size

Our biggest limitation is that we have not obtained a sample big enough for a quantitative analysis. The ideal sample should contain a balanced number of PRs from contributors of different genders and a balanced number of PRs that are merged or non-merged (closed or abandoned after a long period of time) as an indicator of success or failure. Because rejection can also be due to the nature of the code submission or other various reasons, we need to have a large sample to let linguistic patterns manifest.

5.3 Future plan

While writing this submission, we are running a program mining a large corpus of PRs for us to perform quantitative analysis. Once we have the data ready, we will operationalize the linguistic features provided in Table 1 as well as some widely used NLP techniques, such as sentiment analysis [20], politeness strategies [10], and toxicity analysis.[1] The output variable will be whether the PR was merged, indicating a successful contribution and collaboration. From the model, we plan to find if there are certain patterns that more often appear among successful PRs *vs.* non-successful PRs. Then we plan to look at if certain linguistic patterns associated with higher failure rates are more likely to be used among one gender than the other. If so, it might suggest the presence of implicit bias in terms of language use. Because this is an early stage of our big research agenda, we are open to suggestions for refining our method.

REFERENCES

[1] [n. d.]. ([n. d.]).

[2] Toufique Ahmed, Amiangshu Bosu, Anindya Iqbal, and Shahram Rahimi. 2017. SentiCR: A customized sentiment analysis tool for code review interactions. In *International Conference on Automated Software Engineering (ASE)*. IEEE, 106–111.

[3] Penelope Brown, Stephen C Levinson, and Stephen C Levinson. 1987. *Politeness: Some universals in language usage*. Vol. 4. Cambridge university press.

[4] Moira Burke and Robert Kraut. 2008. Mind your Ps and Qs: the impact of politeness and rudeness in online communities. In *Proceedings of the 2008 ACM conference on Computer supported cooperative work*. 281–284.

[5] Fabio Calefato, Filippo Lanubile, Federico Maiorano, and Nicole Novielli. 2018. Sentiment polarity detection for software development. *Empirical Software Engineering* 23, 3 (2018), 1352–1382.

[6] Hilary Carter and Jessica Groopman. 2021. The Linux Foundation Report on Diversity, Equity, and Inclusion in Open Source. https://www.linuxfoundation.org/tools/the-2021-linux-foundation-report-on-diversity-equity-and-inclusion-in-open-source/. Accessed: 2022-03-10.

[7] Gemma Catolino, Fabio Palomba, Damian A Tamburri, Alexander Serebrenik, and Filomena Ferrucci. 2019. Gender diversity and women in software teams: How do they affect community smells?. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS)*. IEEE, 11–20.

[8] Zhenpeng Chen, Xuan Lu, Sheng Shen, Wei Ai, Xuanzhe Liu, and Qiaozhu Mei. 2017. Through a gender lens: An empirical study of emoji usage over large-scale android users. *arXiv preprint arXiv:1705.05546* (2017).

[9] Cristian Danescu-Niculescu-Mizil, Lillian Lee, Bo Pang, and Jon Kleinberg. 2012. Echoes of power: Language effects and power differences in social interaction. In *Proceedings of the 21st international conference on World Wide Web*. 699–708.

[10] Cristian Danescu-Niculescu-Mizil, Moritz Sudhof, Dan Jurafsky, Jure Leskovec, and Christopher Potts. 2013. A computational approach to politeness with application to social factors. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, Vol. 1. 250–259.

[11] Eli Dresner and Susan C Herring. 2010. Functions of the nonverbal in CMC: Emoticons and illocutionary force. *Communication theory* 20, 3 (2010), 249–268.

[12] 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 *2020 IEEE/ACM 42nd International Conference on Software Engineering (ICSE)*. IEEE, 174–185.

[13] Yiru Fangl, Tun Lu, Peng Zhang, Hansu Gu, and Ning Gu. 2018. Exploring the effect of politeness on user contribution in Q&a sites: A case study of stack overflow. In *2018 IEEE 22nd International Conference on Computer Supported Cooperative Work in Design ((CSCWD))*. IEEE, 713–718.

[14] Daviti Gachechiladze, Filippo Lanubile, Nicole Novielli, and Alexander Serebrenik. 2017. Anger and its direction in collaborative software development. In *2017 IEEE/ACM 39th International Conference on Software Engineering: New Ideas and Emerging Technologies Results Track (ICSE-NIER)*. IEEE, 11–14.

[15] GitHub. 2017. Open Source Survey. https://opensourcesurvey.org/2017/. Accessed: 2022-03-10.

[16] Georgios Gousios. 2013. The GHTorrent dataset and tool suite. In *MSR* (San Francisco, CA, USA). 233–236.

[17] Emitza Guzman, David Azócar, and Yang Li. 2014. Sentiment analysis of commit comments in GitHub: an empirical study. In *Proceedings of the 11th working conference on mining software repositories*. 352–355.

[18] Crystal R Herron. 2021. Inclusive Language Matters: Recommendations for Health Care Providers to Address Implicit Bias and Equitable Health Care. *Journal of Maine Medical Center* 3, 2 (2021), 9.

[19] Nasif Imtiaz, Justin Middleton, Joymallya Chakraborty, Neill Robson, Gina Bai, and Emerson Murphy-Hill. 2019. Investigating the effects of gender bias on GitHub. In *2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE)*. IEEE, 700–711.

[20] Bin Lin, Fiorella Zampetti, Gabriele Bavota, Massimiliano Di Penta, Michele Lanza, and Rocco Oliveto. 2018. Sentiment analysis for software engineering: How far can we go?. In *Proceedings of the 40th International Conference on Software Engineering*. 94–104.

[21] Anthony Mulac, James J Bradac, and Pamela Gibbons. 2001. Empirical support for the gender-as-culture hypothesis: An intercultural analysis of male/female

[1] https://www.perspectiveapi.com/

language differences. *Human Communication Research* 27, 1 (2001), 121–152.

[22] Michael J Muller. 2007. *Participatory design: the third space in HCI*. CRC press.

[23] Dawn Nafus. 2012. 'Patches don't have gender': What is not open in open source software. *New Media & Society* 14, 4 (2012), 669–683.

[24] NamSor. [n. d.]. NamSor. https://api.namsor.com/.

[25] Matthew L Newman, Carla J Groom, Lori D Handelman, and James W Pennebaker. 2008. Gender differences in language use: An analysis of 14,000 text samples. *Discourse processes* 45, 3 (2008), 211–236.

[26] Marco Ortu, Giuseppe Destefanis, Steve Counsell, Stephen Swift, Roberto Tonelli, and Michele Marchesi. 2017. How diverse is your team? Investigating gender and nationality diversity in GitHub teams. *Journal of Software Engineering Research and Development* 5, 1 (2017), 1–18.

[27] Bo Pang and Lillian Lee. 2009. Opinion mining and sentiment analysis. *Comput. Linguist* 35, 2 (2009), 311–312.

[28] Rajshakhar Paul, Amiangshu Bosu, and Kazi Zakia Sultana. 2019. Expressions of Sentiments during Code Reviews: Male vs. Female. In *2019 IEEE 26th International Conference on Software Analysis, Evolution and Reengineering (SANER)*. 26–37. https://doi.org/10.1109/SANER.2019.8667987

[29] James W Pennebaker, Martha E Francis, and Roger J Booth. 2001. Linguistic inquiry and word count: LIWC 2001. *Mahway: Lawrence Erlbaum Associates* 71, 2001 (2001), 2001.

[30] Huilian Sophie Qiu, Alexander Nolte, Anita Brown, Alexander Serebrenik, and Bogdan Vasilescu. 2019. Going Farther Together: The Impact of Social Capital on Sustained Participation in Open Source. In *Proceedings of the International Conference on Software Engineering (ICSE)*. IEEE, 688–699.

[31] Huilian Sophie Qiu, Bogdan Vasilescu, Christian K astner, Carolyn Egelman, Ciera Jaspan, and Emerson Murphy-Hill. 2022. Detecting Interpersonal Conflict in Issues and Code Review: Cross Pollinating Open- and Closed-Source Approaches. In *Proceedings of the International Conference on Software Engineering, Software Engineering in Society*.

[32] Dirk Riehle. 2015. How Open Source Is Changing the Software Developer's Career. *Computer* 48, 5 (2015), 51–57.

[33] Jeffrey A Roberts, Il-Horn Hann, and Sandra A Slaughter. 2006. Understanding the motivations, participation, and performance of open source software developers: A longitudinal study of the Apache projects. *Management science* 52, 7 (2006), 984–999.

[34] Daniel Russo and Klaas-Jan Stol. 2020. Gender differences in personality traits of software engineers. *IEEE Transactions on Software Engineering* (2020).

[35] Lucía Santamaría and Helena Mihaljević. 2018. Comparison and benchmark of name-to-gender inference services. *PeerJ Computer Science* 4 (2018), e156.

[36] H Andrew Schwartz, Johannes C Eichstaedt, Margaret L Kern, Lukasz Dziurzynski, Stephanie M Ramones, Megha Agrawal, Achal Shah, Michal Kosinski, David Stillwell, Martin EP Seligman, et al. 2013. Personality, gender, and age in the language of social media: The open-vocabulary approach. *PloS one* 8, 9 (2013), e73791.

[37] Paul Sebo. 2021. Performance of gender detection tools: a comparative study of name-to-gender inference services. *Journal of the Medical Library Association: JMLA* 109, 3 (2021), 414.

[38] Damian A Tamburri, Philippe Kruchten, Patricia Lago, and Hans Van Vliet. 2015. Social debt in software engineering: insights from industry. *Journal of Internet Services and Applications* 6, 1 (2015), 1–17.

[39] Josh Terrell, Andrew Kofink, Justin Middleton, Clarissa Rainear, Emerson Murphy-Hill, Chris Parnin, and Jon Stallings. 2017. Gender differences and bias in open source: Pull request acceptance of women versus men. *PeerJ Comp Sci* 3 (2017), e111.

[40] Bogdan Vasilescu, Daryl Posnett, Baishakhi Ray, Mark G. J. van den Brand, Alexander Serebrenik, Premkumar Devanbu, and Vladimir Filkov. 2015. Gender and tenure diversity in GitHub teams. In *CHI*. 3789–3798.

[41] Zhendong Wang, Yi Wang, and David Redmiles. 2018. Competence-confidence gap: A threat to female developers' contribution on github. In *2018 IEEE/ACM 40th International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS)*. IEEE, 81–90.

[42] Joan C Williams and Rachel Dempsey. 2014. *What works for women at work*. New York University Press.

[43] Katherine J Wolsiefer, Matthias Mehl, Gordon B Moskowitz, Colleen K Cagno, Colin A Zestcott, Alma Tejeda-Padron, and Jeff Stone. 2021. Investigating the Relationship between Resident Physician Implicit Bias and Language Use during a Clinical Encounter with Hispanic Patients. *Health Communication* (2021), 1–9.