

Singapore Management University

## Institutional Knowledge at Singapore Management University

---

Research Collection School Of Computing and  
Information Systems

School of Computing and Information Systems

---

3-2024

### Application of collaborative learning paradigms within software engineering education: A systematic mapping study

Rita GARCIA

Christoph TREUDE

Singapore Management University, [ctreude@smu.edu.sg](mailto:ctreude@smu.edu.sg)

Andrew VALENTINE

Follow this and additional works at: [https://ink.library.smu.edu.sg/sis\\_research](https://ink.library.smu.edu.sg/sis_research)



Part of the [Software Engineering Commons](#)

---

#### Citation

GARCIA, Rita; TREUDE, Christoph; and VALENTINE, Andrew. Application of collaborative learning paradigms within software engineering education: A systematic mapping study. (2024). *SIGCSE 2024: Proceedings of the 55th ACM Technical Symposium on Computer Science Education, Portland, USA, March 20-23*. 1, 366-372.

Available at: [https://ink.library.smu.edu.sg/sis\\_research/8935](https://ink.library.smu.edu.sg/sis_research/8935)

This Conference Proceeding Article is brought to you for free and open access by the School of Computing and Information Systems at Institutional Knowledge at Singapore Management University. It has been accepted for inclusion in Research Collection School Of Computing and Information Systems by an authorized administrator of Institutional Knowledge at Singapore Management University. For more information, please email [cherylids@smu.edu.sg](mailto:cherylids@smu.edu.sg).

# Towards Understanding the Open Source Interest in Gender-Related GitHub Projects

Rita Garcia  
Victoria University of Wellington  
Wellington, New Zealand  
rita.garcia@vuw.ac.nz

Christoph Treude  
University of Melbourne  
Melbourne, Australia  
christoph.treude@unimelb.edu.au

Wendy La  
University of Adelaide  
Adelaide, Australia  
wendy.la@student.adelaide.edu.au

**Abstract**—The open-source community uses the GitHub platform to exchange and share software applications and services of interest. This paper aims to identify the open-source community’s interest in gender-related projects on GitHub. Our findings create research opportunities and identify resources by the open-source community that promote diversity, equity, and inclusion. We use data mining to identify GitHub projects that focus on gender-related topics. We apply quantitative and qualitative methodologies to examine the projects’ attributes and to classify them within a gender social structure and a gender bias taxonomy. We aim to understand the open-source community’s efforts and interests in gender topics through active projects. In this paper, we report on a preponderance of projects focusing on specific gender topics and identify those with a narrow focus. We examine projects focusing on gender bias and how they address this non-inclusive behaviour. Results show a propensity of GitHub projects focusing on recognising and detecting an individual’s gender and a dearth of projects concentrating on the cultural expectations placed on women and men. In the gender bias domain, the projects mainly focus on occupational biases. These findings raise opportunities to address the limited focus of GitHub on gender-related topics through developing projects that mitigate exclusive behaviours.

**Index Terms**—Gender; open-source community; GitHub projects; data mining

## I. INTRODUCTION

The open-source community uses GitHub to exchange and share software applications and services of interest to them. As of July 2022, GitHub had over 46 million public projects that can be used to report emerging technologies and trends across varying domains.<sup>1</sup> The paper’s motivation is to identify the active gender-related projects by the open-source community on GitHub and report on a preponderance of projects focusing on specific gender topics and identify those topics with limited focus. We are interested in understanding the resources provided on GitHub by the open-source community that promote diversity, equity, and inclusion. We define *gender-related projects* as projects that focus on sexual discrimination, gender bias, and stereotyping, along with feminism, health, family, and sexuality [1]. These topics appear within society’s gender structure, influencing the roles and norms created by society for women and men [2] at different dimensions, such as individual and institutional.

To find active gender-related projects, we use data mining, a technique previously used on GitHub to find information

on the changing landscape for machine learning (ML) and artificial intelligence (AI) [3]. Data mining helped understand librarians’ usage of the GitHub platform [4]. Analysing and reporting on GitHub projects can be done using data pipelines, tools that perform data processing to allow researchers and organisations to understand the collected data better [5]. Prior research has mined GitHub to understand the gender dynamics during software development, such as the gender-biased behaviours during Pull Requests (PRs), a submission process for software changes [6]. We use a pipeline to identify the open-source community’s interests in gender-related topics through projects currently active on GitHub. We use the following research questions to guide our study:

- **RQ1:** *How do the attributes of the gender-related projects compare to other active GitHub projects?*
- **RQ2:** *How do the gender-related projects align within the gender social structure?*
- **RQ3:** *What types of gender bias are actively explored within GitHub projects?*

To answer the research questions, we apply a comparison study that uses quantitative and qualitative methods to analyse gender-related projects collected from a pipeline and compare them to a random selection of active GitHub projects to understand the open-source community’s interests in this domain. The qualitative methods classify all gender-related projects within a gender social structure, helping to identify the concentration of GitHub projects within the gender social structure while opening future opportunities to address those areas with a narrow focus. We then align projects to a taxonomy focusing on gender bias, the conscious or unconscious awareness that influences a person’s behaviour [7] that can promote inequitable and non-inclusive environments. The open-source community might respond to gender bias through open-source services and applications, and we examine how they address this non-inclusive behaviour. To the best of our knowledge, this research is the first attempt to identify and report on the open-source community’s interest in gender-related topics on the GitHub platform. The results show 235 active projects where most focus on recognising and detecting an individual’s gender, with a slight interest in interactional behaviours, the cultural expectations placed on men and women [2]. Within the gender bias domain, most of the projects address occupational

<sup>1</sup><https://github.com/search?q=is:public>

bias, with discrimination having minimal representation.

The rest of the paper is structured as follows: Section II presents the background. Section III describes the research methodology. Section IV reports the results, while Section V presents the implications. Section VI describes the study's threats and concludes with Section VII, discussing future research opportunities from our work.

## II. BACKGROUND

### A. Mining GitHub Projects

Mining Software Repositories (MSR) is a field of research that analyses data from software repositories and reports on the collected data [8]. Prior research has used MSR to examine gender diversity in software teams [9], identify the geolocations of GitHub contributors [10], describe a Pull Development model [10], and present the social features [11] on GitHub. These exemplars demonstrate the diverse research opportunities from mining GitHub.

Technologies and tools are available to mine GitHub, such as the GitHub REST API used to create applications that extract data.<sup>2</sup> GHTorrent is a project that uses the GitHub REST API to collect, timestamp, and archive public projects.<sup>3</sup> Like GHTorrent, GitHub Archive archives data hourly, making the data accessible via HTTP.<sup>4</sup> Mining technologies can extract GitHub data, but additional work is required for interpretation since collected data can include biased results [12], such as personal projects [13]. Avoidance strategies can identify active projects that contain a balance of commits, Pull Requests, and stars [13]. Stars represent interested users watching projects through update notifications [14].

There are tools and techniques to mine projects. CLAN (Closely reLated ApplicationNs) [15] is a technique that identifies similarities between Java projects through API method calls. A tool, *reaper* [16], identifies engineered software projects on GitHub. These projects leverage “sound software engineering practices in one or more of its dimensions such as documentation, testing, and project management” [16, p. 3222]. The *reaper* tool removes biased results, such as assignment projects, and identified 24.07% of the 1,857,423 queried projects as engineered software projects. Frameworks support pipeline construction, like PyDriller [17], a Python-based framework that reduces the complexity of mining projects but does not have features for presenting the findings. The tools, techniques, and mining approaches presented in this section provide researchers with various methods to construct pipelines for finding and reporting on extracted GitHub data. However, these approaches mainly focus on programming-related projects and our interests include software development projects and gender-related projects residing in other application domains, such as gender policy.

### B. Gender-Related Research in the Open-Source Community

We review the literature on gender-related topics within the open-source community to understand better how our work resides in this research area. One research area examines gender diversity in software teams, demonstrating women are under-represented as project contributors [18]. A 2019 study [19] showed that 10% of the GitHub project contributors were women, with 2.3% comprising the core developers [20]. This gender imbalance within teams can influence software products. For example, the under-representation of female developers within the FLOSS community has resulted in more unfriendly software towards women [21]. Equal gender representation in software teams can bring together different perspectives and approaches to the development process; however, “the software development profession does not reflect the people who use technology” [22, p. 20]. Gender diversity within teams can promote higher team performance [23], helping to reduce *community smells* [24] and the sub-optimal conditions within the software development communities [25]. Diverse teams can handle issues well and resolve disagreements quickly [26], leading to higher team performance. A study [27] examining software commits showed diverse teams were more productive, concluding gender diversity can positively influence team productivity. To promote team diversity, Vasilescu et al. [27] suggest educational and professional incentives, such as training and outreach programs, to increase the presence of female contributors.

Another research area examines causes for the under-representation of women in open source. Barriers, such as sexism and inequitable pay [28], challenge gender diversity and are a form of gender bias previously observed during the Pull Requests (PRs) process. PRs are source code changes that project maintainers review and comment on before acceptance [29]. Terrell et al. [6] conducted a study that demonstrated when women's gender was identifiable during PRs, their requests were more likely to be rejected than male contributors. Another study by Imtiaz et al. [30] used a framework [31] to examine contributors' GitHub PRs. The framework describes workplace patterns that influence women's career advancements. This study concluded that women concentrated their efforts on smaller sections of projects and a smaller group of organisations. Another study [32] surveyed FLOSS contributors to collect their awareness of gender bias in the community, showing contributors' awareness of gender bias during software collaboration. Still, some contributors “did not see the purpose of attempting to be inclusive, expressing that a discussion of gender has no place in FLOSS” [32, p. 677]. Lee and Carver [32] suggest a zero-tolerance policy for sexist behaviours to reduce challenges and make open source more inclusive. Women in the software industry recommend establishing role models and mentoring and creating inclusive events and groups to mitigate barriers [28].

The literature presented in this section examines gender disparity and bias women encounter in the open-source community. We reviewed literature that uses mining techniques on

<sup>2</sup><https://docs.github.com/en/rest>

<sup>3</sup><https://ghtorrent.org>

<sup>4</sup><https://www.gharchive.org>

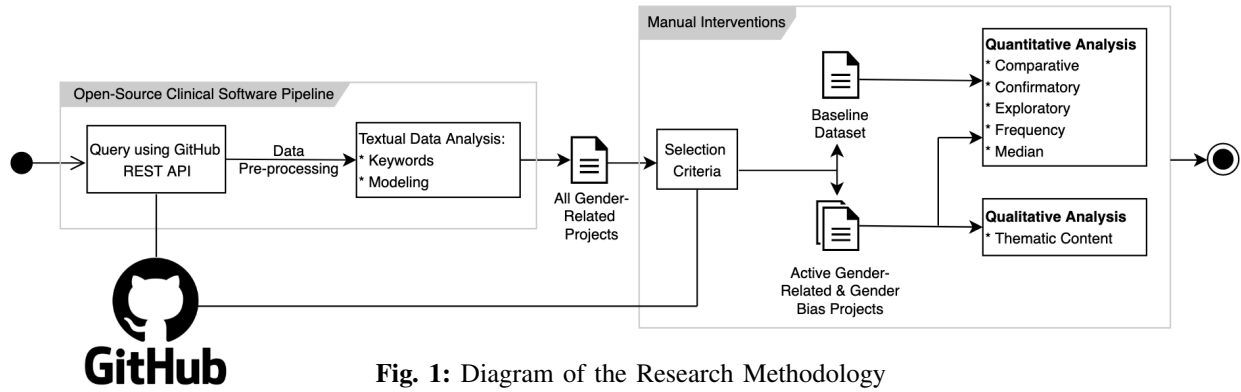


Fig. 1: Diagram of the Research Methodology

open-source projects to understand the diversity and dynamics in software teams. However, from reviewing the literature and to the best of our knowledge, our work is the first attempt to identify the open-source community’s interests in gender-related topics through open-source projects.

### III. RESEARCH METHODOLOGY

We use a comparison study [33], applying quantitative and qualitative methods to evaluate gender-related projects. Figure 1 presents an overview of how the data is collected (See Section III-A) and active projects selected (See Section III-B). Section III-C describes the quantitative analysis that compares attributes of the active gender-related projects with a random selection of active GitHub projects, indicating the level of interest in the gender-related projects. We refer to the random selection of the active GitHub projects as the *baseline dataset* in Figure 1. The qualitative analysis (See Section III-D) provides an in-depth view of the active GitHub projects. The comma-separated values (CSV) files discussed in this section are in our online appendix hosted on Figshare.<sup>5</sup>

#### A. Data Pipeline

We used a Python-based pipeline (See Figure 1, “Open-Source Clinical Software Project”) by Shen and Spruit [34] that identifies open-source clinical software projects for reuse in the healthcare industry. This pipeline was selected for its extraction and analysis features, allowing us to mine GitHub projects within a different domain. Though other pipelines exist, such as a pipeline to identify Internet of Things (IoT) projects on the open web [35] and a Validation Pipeline [36] that evaluates YAML annotated software projects, these pipelines do not report the data necessary for this study’s qualitative analysis (See Section III-D).

The pipeline uses the GitHub REST API to collect data, querying projects’ “name, description, README files, stars, forks, the number of contributors, and the number of commits” [34, p. 3]. The pipeline provides the query as a string which we modified to “gender”, identifying English-based projects created over ten years (2012-2021). The query results were saved as a CSV file, storing the projects’ 17 attributes as

TABLE I: Extracted Data Described by Shen and Spruit [34]

Data Extracted	Description
Numerical Data Types	
Contributors	The number of project contributors.
Creation Date	The project’s creation date.
Forks	The number of project copies. Developers use forks for project contributions.
ID*	The project’s unique key.
Issues	Number of open issues for the project.
README Size	The size of the project’s README in KB.
Source Code Size	Total project size (KB), including its history.
Stars	The number of bookmarks for the project.
Update Date	The latest date the project was updated.
Textual Data Types	
Description	A brief explanation of the project.
Full Name*	Project’s full name.
Languages	Programming languages used in the project.
Owner Location	Location of an owner.
Owner Type	Type of owner: Individual or Organisation.
README URL	URL to the project’s README file.
URL*	API URL providing project information.

\* Data Not Described by Shen and Spruit [34]

separate columns. Table I presents the attributes, organising them alphabetically within their data types: numerical and textual.

We examined the CSV file for data corruption that would impact our analysis and found two projects with corrupted data in the *Description* field. Then, we used the pipeline for textual analysis on the *Description* field to identify informative keywords. IBM-Watson was used for the textual analysis, saving the results to a CSV file for qualitative analysis (See Section III-D) to classify the projects.<sup>6</sup>

#### B. Selection Criteria for Active Projects

We developed the selection criteria to ensure we analysed valid projects and removed biased results, such as students’ homework and dormant projects. The selection criteria select active projects with recent updates, commits, and Pull Requests. We applied recommendations by Kalliamvakou et al.

<sup>5</sup><https://doi.org/10.6084/m9.figshare.19619034>

<sup>6</sup><https://pypi.org/project/ibm-watson>

[13] to construct the selection criteria considering nine perils to avoid biased results in findings. Table II shows the four perils we used and how we applied them to identify active projects. We did not consider the five perils because they focus on software-based projects—for example, *Peril IV. A large portion of repositories is not for software development* [13]. These five perils may exclude projects from our research that do not require programming languages, such as those specialising in gender policy.

**TABLE II:** Selection Criteria for Active Projects

Peril	Selection Rule
A repository is not necessarily a project.	Select projects with three or more forks.
Most projects have very few commits.	Select projects with updates, commits, or pull requests in the last six months (2021-08-01 to 2022-02-01).
Most projects are inactive.	
Two thirds of projects are personal.	Select projects with one or more stars.
Most projects are inactive.	Select projects without “ <i>deprecated</i> ”, “ <i>no longer maintained</i> ”, “ <i>out-of-date</i> ”, and “ <i>unsupported</i> ” in the description field.

We converted these perils into selection rules, shown in Table II. For example, we addressed *Peril III. Most projects are inactive* by selecting projects that demonstrate activity, such as commits, forks, and updates within six months (2021-08-01 to 2022-02-01). Six months was the duration because Kalliamvakou et al. [13] identified that 54% of the GitHub projects were active during this period through commits and Pull Requests. We included project updates to the selection criteria because non-software projects might not use Pull Requests to make changes. The selection criteria also filtered dormant projects using a manual intervention similar to Tang et al. [37] that used the keywords “*deprecated*”, “*no longer maintained*”, “*out-of-date*”, and “*unsupported*” to identify unsupported APIs in applications. In our study, we searched for the exact keywords in the projects’ *Description* field to exclude dormant projects. The selection process applied the selection rules in Table II serially, which generated a CSV file with all active gender-related projects. We describe the analysis further in this section.

### C. Quantitative Analysis

We used quantitative analysis to compare two datasets: the active gender-related projects and a baseline dataset containing a random selection of active GitHub projects. We construct the baseline dataset through random selection, a process previously used by Eaton [4] to create a baseline with GitHub users. But for our baseline, we randomly selected active projects using the selection criteria described in Section III-B and reviewed their descriptions to confirm they did not contain gender-related projects.

We used exploratory analysis to compare the projects’ user types, forks, and stars. We selected these attributes because they provide a view into the open-source community’s involvement (user type and forks) and interest (stars) in these projects.

Exploratory analysis was previously applied to GitHub data, evaluating automotive projects [38] and examining the behaviours of librarians on GitHub [4]. The exploratory analysis was conducted on the two datasets in Excel, performing median analysis of the projects’ user types, forks, and stars.

Confirmatory analysis was used to determine the statistical significance of the gender-related projects. Pearson’s chi-squared test ( $\chi^2$ ) was applied to the two datasets’ user types. We also used two-sample t-tests to compare differences in the projects’ forks and stars within the two datasets.

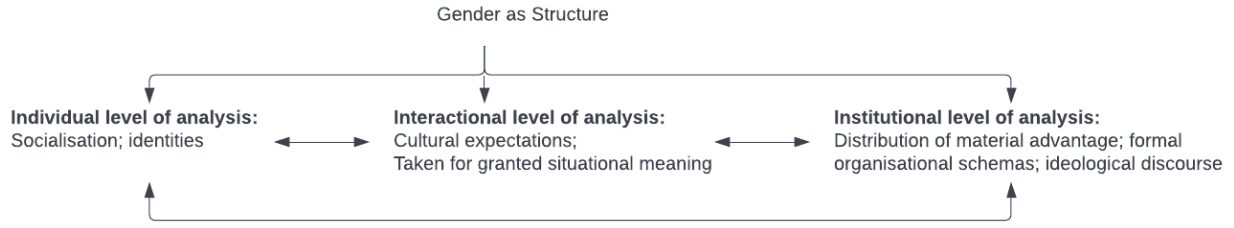
Comparative analysis was applied in this study to determine and compare the application domains used in the projects within the two datasets. This analysis provides a view into how the open-source community addresses gender-related topics using GitHub. Zanartu et al. [39] designed a categorisation tool to identify open-source projects’ application domains by processing their information using machine learning and natural language processing. The tool maps the projects across five application domains classified by Hudson et al. [40]. These application domains are *Application & System Software* that represents end-user systems, *Documentation*, *Non-Web Libraries & Frameworks* that support the construction of non-web-based software applications and services, *Software Tools* that support engineers in the development of software applications and services, and *Web Libraries & Frameworks* that support the construction of web-based software applications and services. The categorisation tool has a command-line interface that takes the projects’ names as a CSV file and produces another CSV file containing the projects’ application domains. After identifying the application domains, we performed frequency analysis in the Excel spreadsheets to create a distribution of application domains in the two datasets.

We also used frequency analysis to evaluate the projects’ programming languages. We identified the distribution of programming languages using a spreadsheet to compare the two datasets and to determine any differences in the languages used within these datasets. We performed the Mann-Whitney U Test [41] using IBM SPSS Statistics v25 to help identify any statistical differences between the projects’ programming languages between the two datasets.

### D. Qualitative Analysis

Qualitative analysis was used to classify the gender-related projects twofold. Section III-D1 describes the first classification process, while Section III-D2 describes the second classification of projects focusing on gender bias.

1) *Classification of All Gender-Related Projects:* We classified the active gender-related projects using thematic content analysis [42]. For the initial coding framework, we used a gender-social structure framework by Risman and Davis [2] to help identify the concentration of projects within society’s gender structure. The framework describes the roles and norms created by society for women and men within the social structure. Figure 2 shows the framework’s three dimensions. The *Individual* dimension focuses on the progress of gender identity and how the individual socialises. The *Interactional*



**Fig. 2:** Gender Social Structure Framework [2]

dimension focuses on the cultural expectations for women and men and the presumptions placed on them even when performing equivalent societal roles. The *Institutional* dimension examines how culture influences actions through regulations and gender-specific organisational schemas. Risman and Davis posit the framework's dimensions have a cyclic relationship, influencing the dynamics in the social structure that can contribute to society's gender inequalities.

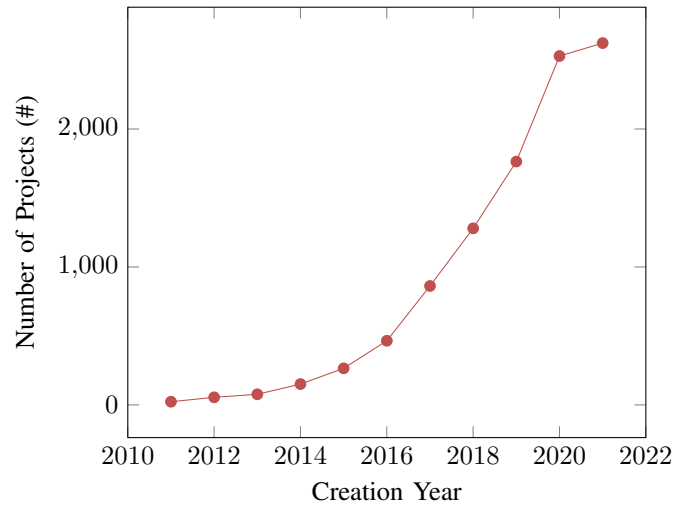
The initial coding framework contained four nodes: the three gender social structure dimensions and *Unrelated* to represent projects unrelated to the gender topics. The pipeline extracted the projects' informative keywords (See Section III-A) to classify the projects into four nodes. The qualitative process marked any active project unrelated to gender into the *Unrelated* node. The primary author performed the coding process with the projects' informative keywords. For coding reliability, the authors met to discuss the framework's dimensions to agree on the coding process. The meeting helped establish the accuracy of the coding process by confirming the authors' understanding of how the projects' keywords relate to the framework's dimensions. The authors decided to group "gender detection" and "gender prediction" keywords and place them within the *Individual* dimension because the projects use the information to identify an individual's gender. The projects were re-coded after the discussion by one author to ensure the agreed-upon coding process was accurately applied to the projects.

Because the projects contained a variety of informative keywords with similar meanings or goals, such as "Female Gender Classifier" and "gender classifier", we used NVivo, a qualitative analysis tool, to consolidate similar keywords and phrases into groups. We used these groups to classify the projects into the framework's dimensions. If projects did not have informative keywords, we evaluated the READMEs to understand their purpose. Reviewing the README files was performed on 18.72% of the projects exclusively using the keyword "gender", a broad term that lacks details on the project's purpose. During the classification process, we excluded project keywords (9.36%) that did not relate to gender, such as "Javascript" and "Must-read papers". If all the project's keywords were not gender-related, we placed the project into the *Miscellaneous* group within the *Unrelated* category. For projects (n=60, 20.34%) containing multiple keywords related to gender, we coded them into more than one dimension. For example, the Inclusive Design Toolkit [43] is

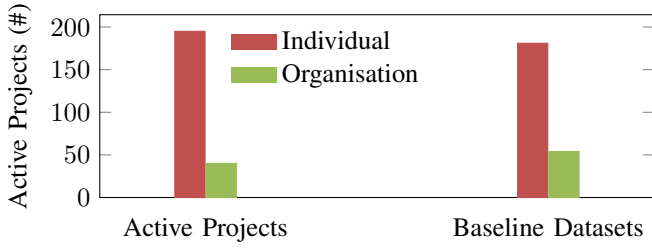
a project that promotes diversity and inclusion. This project was placed in the *Inclusion* and *Income* groups and classified within the *Institutional* and *Individual* dimensions. The coding process formed a matrix we extracted from NVivo to present the findings.

2) *Classification of Gender Bias Projects:* We conducted the second classification after the gender-related projects were classified (See Section III-D1), involving the projects from the *Gender Bias* group. We extracted these projects for thematic context analysis using a taxonomy by Doughman et al. [44] designed to help identify gender biases and stereotypes in the English language. This taxonomy was formed through observations on artificial intelligence, machine learning, and natural language processing, propagating and reinforcing prejudices when biased text is integrated into these systems. Table VI presents the five gender bias types in the taxonomy, along with the types' definitions. These five types are *Generic Pronouns*, *Semantics*, *Sexism*, and *Exclusionary and Occupational Bias*. Each type contains subtypes. For example, the *Sexism* type has *Hostile* and *Benevolent* sexism subtypes.

This classification process involved two authors to ensure coding reliability. The classification required inspecting multiple areas of the project to understand its purpose, such as source code. Using an Excel spreadsheet, the authors separately coded all the projects. After coding, the authors



**Fig. 3:** Gender-Related Projects Cumulated by Creation Year



**Fig. 4:** Comparison of Projects' User Types

met to agree upon the coding criteria, which included a discussion of the discrepancies from the initial coding. The authors discussed their interpretations of the *Exclusionary Bias* type and the subtypes within the *Generic Pronouns* and *Semantics*. The authors decided to apply *Generic Pronouns* when masculine representation was assumed, while *Semantics* was selected when addressing sexist stereotypes. The authors also negotiated emerging types, agreeing on the types and their labels. After the negotiation, one author re-coded all the gender bias projects using the agreed-upon coding criteria. Re-coding also included reviewing the projects' README files, artefacts, and any associated publications identified within the README files. Upon completion of the re-coding, the results were quantified in the spreadsheet.

#### IV. RESULTS AND DISCUSSION

The pipeline's *gender* query string identified 10,094 gender-related GitHub projects created over the ten years (2011-2021). Figure 3 displays these projects by creation year, showing the number of projects increasing over the subsequent years. The query results show that 2017 to 2020 had over 400 newly created projects, with 2016 (n=397) approaching this figure. A potential reason for the increased number of projects using the term *gender* might be the resurgence of the MeToo movement in 2017 [45]. This social movement raises awareness about sexual violence to let survivors know they are not alone. Though the movement was initiated in 2006 by Tarana Burke [46], the Twitter #MeToo hashtag used by actress Alyssa Milano in October 2017 might have promoted interest within the digital space.

The selection criteria identified 235 (2.31%) active gender-related projects from the 10,094 projects. The exclusion process removed 9661 projects with less than three forks, 160 projects without activity for six months, and 38 with zero stars. In this section, we further discuss the active gender-related projects by answering the paper's three research questions.

**A. RQ1: How do the attributes of the gender-related projects compare to other active GitHub projects?**

We compare the projects' attributes to understand how the 235 active gender-related projects align with a random selection of active GitHub projects. Figure 4 presents the comparison of the owner types. The table shows a similar composition of users creating gender-related projects and the random selection of active GitHub projects. The results show

**TABLE III:** Comparison of Projects' Forks and Stars

Datasets	Attributes ( $\bar{x}$ )	
	Forks	Stars
Gender-Related	7	20
Baseline	44	316
$\chi^2$ , $p$ -value	$\chi^2 = -4.050$ , $p < .00003$	$\chi^2 = 9.072$ , $p < .00001$

The result is significant at  $p < .05$ .

the majority (Active=82.98%, Baseline=77.02%) of the users are *Individuals*, a term used by GitHub to differentiate projects owned by organisations and personal accounts. Our results are supported by Pearson's chi-squared test ( $\chi^2 = 2.6064$ ,  $p$ -value = .106434) with a moderate effect size ( $p \leq 0.05$ ) that showed no statistical differences between the datasets' user types.

We also compared the projects' forks and stars attributes. Table III shows a median value of 7 for the active gender-related projects' forks, while the median value for the baseline dataset projects is 44. Comparing the projects' stars also demonstrates similar results. There is a lower number (median=20) of stars for the active gender-related projects compared to the higher median value of 316 for the projects in the baseline dataset. The statistical significance supports these results in the forks ( $t(235)_{Forks} = -4.05011$ ,  $p < .00003$ ) and stars ( $t(235)_{Stars} = 9.07231$ ,  $p < .00001$ ) measured with moderate (.05) effect size. These results potentially demonstrate similar user types creating gender-related projects, but more work is required to raise awareness and interest within the open-source community about gender-related projects.

Results from the application domains provide insight into the types of projects created. Table IV shows the *Non-Web Libraries & Frameworks* application domain is the most prominent (n=166, 70.64%) within the active gender-related projects. In contrast, the random selection of GitHub projects in the baseline dataset focused on the *Web Libraries & Framework* application domain (n=94, 40.00%). Both datasets show limited focus on the *Application & System Software* application domain, where the active gender-related projects had seven (2.98%) and nine (3.83%) projects in the baseline dataset. A potential reason for these findings might be that gender-related projects contain task-based scripts, such as automated gender detection. The gender problem domain might not suit the solutions presented as stand-alone applications.

Figure 5 compares the top-10 programming languages used by the gender-related projects and the baseline dataset. The results show Python (n=101, 45.29%) was predominately used by the active gender-related projects, while the random selec-

**TABLE IV:** Comparison of Projects' Application Domains

Domain	All Gender	Baseline
Application & System Software	7 (2.98%)	9 (3.83%)
Documentation	34 (14.47%)	16 (6.81%)
Non-Web Libraries & Frameworks	166 (70.64%)	72 (30.64%)
Software Tools	8 (3.40%)	44 (18.72%)
Web Libraries & Frameworks	20 (8.51%)	94 (40.00%)



tion of active GitHub projects predominately used JavaScript (n=92, 39.15%). The Mann-Whitney U Tests ( $p < 0.05$  two-tailed) show significant differences for the *JavaScript*, *Jupyter Notebook*, and *Python* languages when comparing the two datasets. More research is required to understand the different applications of languages in gender-related projects.

**RQ1 Summary:** The active gender-related projects have a similar composition of project owners as a random selection of active GitHub projects; however, the forks and stars results show lower interest by the open-source community in these gender-related projects. When comparing the two datasets, we found differences in the programming languages used by the projects, but both have limited representation within *Application & System Software* application domain. Work is required to understand the application scenarios for gender-related projects and how to promote interest in these projects within the open-source community.

*B. RQ2: How do the active gender-related GitHub projects align within the gender social structure?*

We classified 235 active projects within the gender social structure framework [2], shown in Table V. The table organises the findings within the three gender social structure dimensions and an *Unrelated* category in descending order. We used the projects’ informative keywords for the coding process, but we also evaluated 71 (30.71%) projects’ README files to understand their purpose better. Within these 71 projects, 44 (18.72%) used the keyword “gender”, 22 projects (9.36%) contained only keywords unrelated to gender, such as “*R package*” and “*Chrome extension*”, and five (2.13%) projects required further analysis because they did not have keywords. As a result, 25 (8.47%) projects were placed in the *Unrelated* category to represent projects that could not be classified within the gender social structure.

Table V shows the coding results with 16 project groups across four categories. The table organises the groups in descending order within the categories, providing group descriptions and an exemplar project as an illustration. For example, the *Gender Diversity* group resides in the *Institutional* (n=32, 10.85%) dimension and contains seven (2.37%) projects.

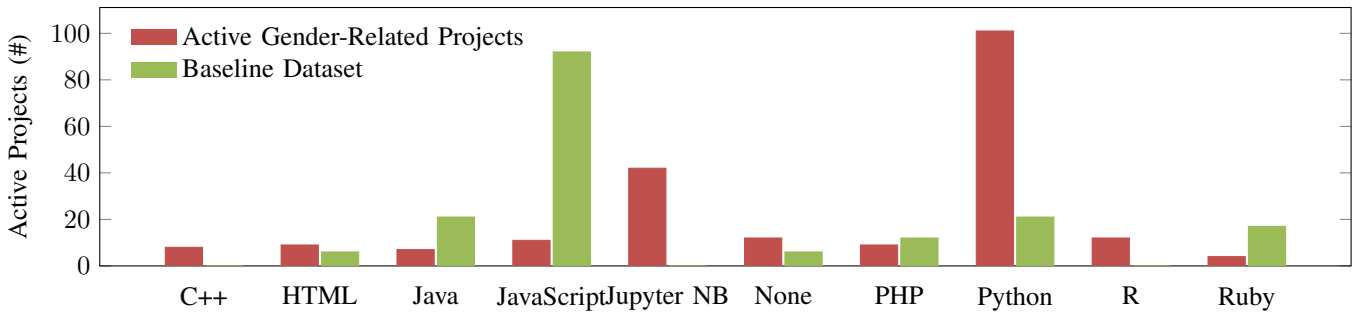
The table contains 295 nodes to represent the coded projects. Some (n=98, 41.70%) projects included more than one keyword, demonstrating their varying interests. Sixty

(20.34%) projects with more than one keyword were classified in multiple gender dimensions. For example, the project by Van Campenhout [55] provides an algorithm used in a study [56] that examines gender homophily — the preference a person has for interacting with others of the same gender [61] — during the training of farmers in Uganda. The study investigates gender composition and female and male trainers’ roles in acquiring knowledge from farmers. The project contained the keywords “*Gender Role*” and “*Gender Composition*”, placing it in the two groups with the same names and classified within the gender social structure’s *Interactional* and *Institutional* dimensions.

The *Individual* dimension contains the most (n=219, 74.24%) projects. The majority (n=165, 55.59%) are concentrated within the *Detectors and Predictors* group. This group focuses on the detection and prediction of an individual’s gender by using a person’s name or by extracting information from audio and image files. For example, FaceLib [47] is a Python-based library that recognises an individual’s facial expressions and gender through image files. Other projects in the *Individual* dimension use an individual’s attributes for *Gender Classification* (n=20, 6.78%), *Income* (n=3, 1.02%), and *Sexual Identity and Orientation* (n=3, 1.02%).

The *Interactional* (n=19, 6.44%) dimension, focusing on the cultural expectations placed on women and men, contains the least number of projects. For example, some projects concentrate on *Gender Bias* (n=18, 6.10%) that analyse gender-biased language. For instance, gender-bias [60] is a project that contains a Python-based tool designed to identify and report gender-biased language in recommendation letters for the medical professions. Also, within the *Gender Bias* group, a project [62] uses its README to list URL links to gender-related conferences and journal publications. The *Gender Role* (n=1, 0.34%) group contains one project [55] that focuses on society’s expectations of how the genders behave, communicate, and conduct themselves. This previously mentioned project supports a study [56] that examines how gender plays a role in the knowledge acquisition of Ugandan maize farmers educated by female and male trainers.

The *Institutional* (n=32, 10.85%) dimension projects examine how culture influences gender-specific actions that emerge from regulations and organisational schemas. These include *Gender Disparity* (n=14, 4.75%) and *Gender Diversity*



**Fig. 5:** Comparison of Projects’ Programming Languages



**TABLE V:** Classification of Active GitHub Projects within the Gender Social Structure Framework

Group	Total (%)	Project Descriptions	Example Project
<b>Individual (n=219, 74.24%)</b>			
Detectors and Predictors	164 (55.59%)	Detects faces and emotions using age and gender; Uses facial attributes to estimate age and gender; Trains ML models to detect gender; Detects gender, race, and emotion in real-time.	FaceLib [47] is a Python-based library that recognises facial expressions and gender using an image file of an individual.
Gender of a Name	25 (8.47%)	Assists in legal name changes; Guesses and trains models to detect gender from names.	randomNames [48] contains a function that randomly selects names based on gender as the input.
Gender Classification	20 (6.78%)	Classifies speakers gender using names, voices, and images.	Speaker Gender Classification [49] is a deep learning Jupyter Notebook project that classifies a person's gender based on an audio sample.
Income	3 (1.02%)	Assists in inclusive designs for bank services and home loan applications.	Inclusive Design Toolkit [43] provides documentation from the Government of Ontario, promoting diversity through services like income.
Sexual Identity and Orientation	3 (1.02%)	Creates an ontology for sexual orientation and gender.	Gender, Sex, and Sexual Orientation (GSSO) [50] is a web-based project for gender-related topics, such as sexual identity, orientation, and behaviour.
Social Media	3 (1.02%)	Profiles social media authors; Constructs social media followers.	A model, gender_prediction [51], within Jupyter Notebook that uses social media profiles to predict gender.
Analysis of Gender Splits	1 (0.34%)	Analyses of gender distribution among TV and radio hosts and panel members.	The UK Panel Show Gender Breakdown [52] provides statistics of panel guests and hosts on UK TV shows.
<b>Institutional (n=32, 10.85%)</b>			
Gender Disparity	14 (4.75%)	Contains communities promoting equality in Tech; Analyses citations for gender balance; Promotes equality in legal representation.	PyLadies [53] is a project owned by the PyLadies organisation that supports women in becoming active participants in the open-source community.
Inclusion	9 (3.05%)	Detects gender-inclusive language; Provides suggestions for mitigating exclusive language; Contains datasets on hate speech and gender-inclusive language.	Inclusive Design Toolkit [43] provides documentation and presentations from the Government of Ontario, Canada that promotes diversity through services, such as income and culture.
Gender Diversity	7 (2.37%)	Promotes gender diversity in gaming and language; Contains a gender-balanced dataset for Artificial Intelligence.	A project by GameHer [54], a French association promoting gender diversity in the gaming community using tournaments and inclusive gaming environments.
Gender Composition	2 (0.68%)	Contains population information focusing on gender to promote positive change.	A project [55] that supports a conference paper [56] containing an algorithm that examines the effects of gender homophily in Uganda farmers when learning.
<b>Unrelated (n=25, 8.47%)</b>			
Language	19 (6.44%)	Contains dictionaries, tools, and games on gendered languages and pronouns.	A French dictionary project [57] in CSV format containing pronouns and verbs with gender.
Miscellaneous	4 (1.35%)	Provides ridership dataset for bike-sharing programs; Contains assignments, laser-cut prototyping, and hotel management system.	Citi Bike Analytics Jersey City 2020 [58] examines data to better understand the growth rate of the service, which includes gender distribution of Citi Bike users.
Gender Fields	2 (0.68%)	Provides gender data component for content management systems.	craft3-fields [59] contains custom fields for Craft Content Management System that includes gender.
<b>Interactional (n=19, 6.44%)</b>			
Gender bias	18 (6.10%)	Contains gender bias and debiasing tools; Lists gender bias papers; Analyses of gender bias using Google Translate.	A project, gender-bias [60], is a Python-based tool designed to identify and report gender-biased language in recommendation letters for medical professionals.
Gender Role	1 (0.34%)	Algorithm to support conference paper on gender homophily.	A project [55] containing an algorithm that supports a conference paper [56] that examines the effects of gender homophily for Uganda farmers when learning.

(n=7, 2.37%), which examine and promote gender equity and diversity in Tech. For example, the *Gender Disparity* group contains a PyLadies [53] project for the association's website maintained by their infrastructure team. PyLadies is an association that supports women in becoming active participants in the open-source community. The final category, *Unrelated* (n=25, 8.47%), contains 25 (8.47%) projects that did not fit into the framework. Most (n=19, 6.44%) of these projects focus on *Language*, providing users with educational games and dictionaries. For example, one project [57] contains a French dictionary that contains pronouns and verbs with their gender.

Our results show gender-related projects that promote change and raise awareness of the gender dynamics at the *Institutional* and *Interactional* dimensions. However, there is a limited (n=51, 17.29%) number of projects focusing on these dimensions. A potential reason for these results might be the methods required to instigate change in these dimensions, which includes policies to promote diversity at educational institutions [70] and interventions to mitigate gender bias during the hiring process [71]. These methods to promote change and awareness might prominently use policies and documentation, whereas institutions and organisations might use other platforms and services outside of GitHub for their services. These

**TABLE VI:** Classification of Active Gender Bias GitHub Projects

Types	Subtypes	Total (%)	Definition	Example Project
<b>Gender Bias Taxonomy [44]</b>				
Occupational Bias	<ul style="list-style-type: none"> <li>• Gendered Division of Labour</li> <li>• Gendered Rules &amp; Duties</li> </ul>	13 (33.33%)	Applies stereotypical gendered language to professions, such as referencing a role by a certain gender.	<code>gender-bias-BERT</code> [63] evaluates in English and German the gender bias in various professions.
Generic Pronouns	<ul style="list-style-type: none"> <li>• Generic He</li> <li>• Generic She</li> <li>• Gendered Generic Man</li> </ul>	8 (20.51%)	Uses a generic pronoun with sex-indefinite antecedents.	<code>gender-debias</code> [64] contains a debiasing dataset for ML training. Contains feminine and masculine German pronouns that change due to gender bias.
Semantics	<ul style="list-style-type: none"> <li>• Metaphors</li> <li>• Gendered Attributes</li> <li>• Old Sayings</li> </ul>	8 (20.51%)	Applies words to sentences in a demeaning way that alters their semantic meanings.	<code>gender-bias</code> [60] is a Python-based tool designed to identify and report gender-biased language in recommendation letters for medical professions.
Sexism	<ul style="list-style-type: none"> <li>• Hostile Sexism</li> <li>• Benevolent Sexism</li> </ul>	5 (12.82%)	Applies words to create positive or negative expressions towards women and men depending on the context.	<code>Debiased-Chat</code> [65] trains dialogue models to reduce gender bias, using the models to rephrase the pronouns that promote sexist expressions.
Exclusionary Bias	<ul style="list-style-type: none"> <li>• Explicit Marking of Sex</li> <li>• Gender-based Neologisms</li> <li>• Gendered Word Ordering &amp; Division</li> </ul>	2 (5.13%)	Imposes gender on an unknown or neutral entity.	A project, <code>Double-Hard-Debias</code> [66] supports the debiasing of word embeddings. Contains a dictionary that addresses exclusionary bias.
<b>Emerging Codes</b>				
Imaging Bias		2 (5.13%)	Addresses biases that emerge during AI and ML image detection.	A project [67] that contains a framework that examines and mitigates various biases, including gender, within visual recognition tasks.
Discrimination		1 (2.57%)	Applies prejudicial treatment based on a person's gender or other background.	A project [68] supporting a paper [69] analysing legal criminal cases in India to determine in-group bias within the judicial system.

methods might not require software applications and services to promote and disseminate change, which may explain the smaller representation of these projects on GitHub. However, more work is necessary to draw a conclusion.

**RQ2 Summary:** The majority (n=219, 74.24%) of the active gender-related projects focus on the individual, such as recognising and detecting a person's gender through input files. A small (n=19, 6.44%) group of *Interactional* projects focuses on the cultural expectations placed on women and men, which might require methods like policy to instigate and disseminate change. Organisations might not use GitHub to discuss gender policy, but more work is necessary to understand the limited focus on this dimension.

*C. RQ3: What types of gender bias are actively explored within GitHub projects?*

The coding of all active gender-related projects identified 18 focusing on gender bias. Table VI shows these 18 projects classified in the gender bias taxonomy, organising the results by types followed by the emerging gender bias types identified during the coding process. The 18 projects produced 39 classification nodes, demonstrating that some projects focused on more than one gender bias type. For example, Table VI shows the `gender-bias` [60] project — a tool that examines letters of recommendation for the medical profession — is classified under *Semantics* and *Occupational Bias*.

Table VI presents the coding for the gender bias taxonomy, including two emerging types, *Imaging Bias* (n=2, 5.13%) and *Discrimination* (n=1, 2.57%). These emerging types did not appear in the initial coding framework because the taxonomy focuses on gender bias in the English language. In contrast, *Imaging Bias* focuses on biases during visual recognition tasks, and *Discrimination* concentrates on a person's prejudicial treatment of a group based on their gender or other backgrounds, such as race. An exemplar project [67] for *Imaging Bias* contains a framework that examines and mitigates various biases, including gender, within visual recognition tasks. For *Discrimination*, the `gender-judicial-bias-india` project [68] includes files necessary to replicate a study [69] evaluating the legal criminal cases in India for in-group bias. We observed the majority (n=11, 61.11%) of the gender bias projects supporting research, such as `gender-bias-BERT` [63], that supports a published thesis [72] that evaluates gender bias within different professional groups.

The results show the majority (n=13, 33.33%) of the projects focused on *Occupational Bias*, the stereotypical gendered language applied to societal roles. For example, `gender-bias-BERT` [63] is a project that contains a tool that evaluates the English and German languages for gender bias that exists in various professions, such as medical professions. The results show an equal number of projects (n=8, 20.51%) within the *Generic Pronouns* and *Semantics* types.

An exemplar *Generic Pronouns* project is *gender-debias* [64], which contains feminine and masculine German pronouns that change due to gender bias. An example of a *Semantics* project is the previously mentioned *gender-bias* [60] project that reports on gendered language within letters of recommendation for the medical profession.

The least applied gender bias type is *Exclusionary Bias* (n=2, 5.13%), a bias that assumes and assigns a gender to a neutral or unknown entity. For example, the *Double-Hard-Debias* [66] project focuses on debiasing *word embeddings*, a natural language processing (NLP) concept that analyses text through word representation.

The *Imaging Bias* (n=2, 5.13%) type contains projects considering the stereotypes and inequities that emerge from visualisation tools. However, we observe from our findings a disparity between those projects (n=2) mitigating imaging bias and (n=164) projects focusing on visual recognition through gender (See Section IV-B). Perhaps some 164 projects are considering imaging bias, but additional analysis is necessary to draw a conclusion.

**RQ3 Summary:** We were able to code most (n=15, 83.33%) of the gender bias projects within the taxonomy. Three projects addressed gender bias behaviours within visualisation tools and discrimination. *Occupational Bias* is the most common type, while *Exclusionary Bias* is the least applied. Though the two emerging types contained a small (n=3, 7.70%) number of gender bias projects, it demonstrates the open-source community is considering aspects of gender bias within social behaviours and mitigating bias within the software.

## V. IMPLICATIONS

**To Educators:** We encourage educators to teach the ethical implications of gender detection because our findings show a strong interest in automated methods for identifying an individual’s gender. *Nature Portfolio* surveyed artificial intelligence (AI) experts, collecting their opinions on facial-recognition research [73]. The results showed some experts had ethical concerns in the AI field. Teaching students the implications of unethical projects might help address these concerns. Educators can also help raise students’ awareness of interactional topics, such as gender bias, to potentially get them involved in open-source projects that address these topics. This might motivate students to get involved in ethical practices, such as research labs [74] that use machine and deep learning technologies to identify hate speech. Educators can also minimise students’ exposure to gender-biased software. For example, a checklist exists for K-12 programs [75] for educators to help identify gender biases in software. The checklist can assist educators in making informed decisions on adopting more inclusive software.

**To Open-Source Community:** The open-source community can increase awareness of the gender-related work being conducted in their community. One way to raise awareness is through the GitHub collections, a curated list of projects posted on GitHub that highlight the work being done in

communities and industries on various topics and interests.<sup>7</sup> Showcasing these projects as a collection could encourage more members of the open-source community to get involved in these gender-related projects.

**To Researchers:** The research community may be addressing gender-related topics outside of the GitHub platform in the *Interactional* dimension. We encourage researchers to make available their work on promoting diversity, equity, and inclusion on open-source platforms. The presence of this work on open-source platforms might encourage the community to create and adapt projects to address gender-related topics, especially areas with a narrow focus, like *Imaging Bias*.

## VI. THREATS TO VALIDITY

There are limitations to this research. The first relates to content validity since our study’s reliability depends on the projects identified by the pipeline. We cannot say that the pipeline identified all gender-related projects; however, we used the GitHub REST API to validate the collected data to strengthen the results. Another threat is data consistency. To reduce the margin of variability in the qualitative results, we applied the approaches Silverman [76] suggested to improve reliability. The approaches include constant data comparison, in which the authors discuss the coding criteria, process, and results until agreeing.

## VII. CONCLUSION AND FUTURE WORK

The motivation for this paper is to understand the open-source community’s interest in gender-related topics by evaluating their active projects on GitHub. We used a pipeline to identify the active gender-related GitHub projects, classifying them twofold, first identifying the active projects within a gender social structure and then classifying projects within a gender bias taxonomy. The results show most gender-related projects focus on the individual, performing gender detection and recognition. We observed a limited focus on interactional topics, such as projects addressing gender bias. Within gender bias, these projects primarily focus on occupational bias, with a small representation focusing on discrimination.

Future research can investigate the dearth of interactional projects to understand better the limited focus on detecting gendered behaviours. As Section V mentions, the education, open-source, and research communities can raise awareness of projects focusing on interactional topics, helping promote more project development in this area. There is a possibility GitHub projects focus on interactional topics, but these projects might not specify their intentions through keywords and descriptions. Future research can perform additional queries to identify other gender-related projects and can analyse more projects in-depth to identify their purpose better.

## REFERENCES

- [1] A. S. Runyan and V. S. Peterson, *Global Gender Issues in the New Millennium*. Routledge, 2014.

<sup>7</sup><https://github.com/collections>

- [2] B. J. Rismann and G. Davis, "From sex roles to gender structure," *Current Sociology Review*, vol. 61, no. 5-6, pp. 733-755, 2013.
- [3] D. Gonzalez, T. Zimmermann, and N. Nagappan, "The state of the ml-universe: 10 years of artificial intelligence & machine learning software development on GitHub," in *Proceedings of the 17th International Conference on Mining Software Repositories*, 2020, pp. 431-442.
- [4] M. E. Eaton, "A comparative analysis of the use of GitHub by librarians and non-librarians," *Evidence Based Library and Information Practice*, vol. 13, no. 2, pp. 27-47, 2021.
- [5] A. R. Munappy, J. Bosch, and H. H. Olsson, "Data pipeline management in practice: Challenges and opportunities," in *Product-Focused Software Process Improvement, PROFES 2020. Lecture Notes in Computer Science*. Springer, Cham, 2020, vol. 12562, pp. 168-184.
- [6] J. Terrell, A. Kofink, J. Middleton, C. Rainear, E. Murphy-Hill, C. Parnin, and J. Stallings, "Gender differences and bias in open source: Pull request acceptance of women versus men," *PeerJ Computer Science*, vol. 3, no. e111, 2017.
- [7] Y. Wang and D. Redmiles, "Implicit gender biases in professional software development: An empirical study," *Proceedings of the 41st International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS '19)*, pp. 1-10, 2019.
- [8] K. S. Herzig and A. Zeller, "Mining your own evidence," in *Making Software*. Sebastopol, Calif., USA: O'Reilly, 2011, pp. 517-529.
- [9] G. A. A. Prana, D. Ford, A. Rastogi, D. Lo, R. Purandare, and N. Nagappan, "Including everyone, everywhere: Understanding opportunities and challenges of geographic gender-inclusion in OSS," *IEEE Transactions on Software Engineering*, pp. 1-19, 2021.
- [10] G. Gousios, M. Pinzger, and A. v. Deursen, "An exploratory study of the pull-based software development model," in *Proceedings of the 36th International Conference on Software Engineering*, ser. ICSE 2014, 2014, pp. 345-355.
- [11] A. Begel, J. Bosch, and M.-A. Storey, "Social networking meets software development: Perspectives from GitHub, MSDN, Stack Exchange, and TopCoder," *IEEE Software*, vol. 30, no. 1, pp. 52-66, 2013.
- [12] G. Gousios and D. Spinellis, "Mining software engineering data from GitHub," in *Proceedings of the 39th International Conference on Software Engineering Companion (ICSE-C)*, 2017, pp. 501-502.
- [13] E. Kalliamvakou, G. Gousios, K. Blincoe, L. Singer, D. M. German, and D. Damian, "The promises and perils of mining GitHub," in *Proceedings of the 11th Working Conference on Mining Software Repositories*, ser. MSR 2014, 2014, pp. 92-101.
- [14] Y. Zhang, D. Lo, P. S. Kochhar, X. Xia, Q. Li, and J. Sun, "Detecting similar repositories on GitHub," in *Proceedings of the 24th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, 2017, pp. 13-23.
- [15] C. McMillan, M. Grechanik, and D. Poshvanyk, "Detecting similar software applications," in *Proceedings of the 34th International Conference on Software Engineering*, ser. ICSE '12, 2012, pp. 364-374.
- [16] N. Munaiah, S. Kroh, C. Cabrey, and M. Nagappan, "Curating GitHub for engineered software projects," *Empirical Softw. Engg.*, vol. 22, no. 6, pp. 3219-3253, 2017.
- [17] D. Spadini, M. Aniche, and A. Bacchelli, "Pydriller: Python framework for mining software repositories," in *Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, ser. ESEC/FSE 2018, 2018, pp. 908-911.
- [18] W. E. Powell, D. S. Hunsinger, and B. D. Medlin, "Gender differences within the open source community: An exploratory study," *Journal of Information Technology Management*, vol. 11, no. 4, pp. 29-37, 2010.
- [19] A. Bosu and K. Z. Sultana, "Diversity and inclusion in open source software (OSS) projects: Where do we stand?" in *Proceedings of the ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, 2019, pp. 1-11.
- [20] E. D. Canedo, R. Bonifácio, M. V. Okimoto, A. Serebrenik, G. Pinto, and E. Monteiro, "Work practices and perceptions from women core developers in oss communities," 2020.
- [21] Y. Lin. (2005) Gender dimensions of Floss development. [Online]. Available: <https://www.metamute.org/editorial/articles/gender-dimensions-floss-development>
- [22] K. Albusays, P. Bjorn, L. Dabbish, D. Ford, E. Murphy-Hill, A. Serebrenik, and M.-A. Storey, "The diversity crisis in software development," *IEEE Software*, vol. 38, no. 2, pp. 19-25, 2021.
- [23] S. G. Rogelberg and S. M. Rumery, "Gender diversity, team decision quality, time on task, and interpersonal cohesion," *Small Group Research*, vol. 27, no. 1, pp. 79-90, 1996.
- [24] G. Catolino, F. Palomba, D. A. Tamburri, A. Serebrenik, and F. Ferrucci, "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)*, 2019, pp. 11-20.
- [25] D. A. Tamburri, P. Kruchten, P. Lago, and H. van Vliet, "Social debt in software engineering: Insights from industry," *J Internet Serv Appl*, vol. 6, no. 10, 2015.
- [26] P. C. Earley and E. Mosakowski, "Creating hybrid team cultures: An empirical test of transnational team functioning," *The Academy of Management Journal*, vol. 43, no. 1, pp. 26-49, 2000.
- [27] B. Vasilescu, D. Posnett, B. Ray, M. G. van den Brand, A. Serebrenik, P. Devanbu, and V. Filkov, "Gender and tenure diversity in GitHub teams," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, ser. CHI '15, 2015, pp. 3789-3798.
- [28] B. Trinkenreich, R. Britto, M. A. Gerosa, and I. Steinmacher, "An empirical investigation on the challenges faced by women in the software industry: A case study," in *2022 IEEE/ACM 44th International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS)*, 2022, pp. 24-35.
- [29] O. Kononenko, T. Rose, O. Baysal, M. Godfrey, D. Theisen, and B. de Water, "Studying pull request merges: A case study of Shopify's active merchant," in *Proceedings of the 40th International Conference on Software Engineering: Software Engineering in Practice*, ser. ICSE-SEIP '18, 2018, pp. 124-133.
- [30] N. Imtiaz, J. Middleton, J. Chakraborty, N. Robson, G. Bai, and E. Murphy-Hill, "Investigating the effects of gender bias on GitHub," in *Proceedings of the 41st International Conference on Software Engineering*, ser. ICSE '19. IEEE Press, 2019, p. 700-711.
- [31] J. Williams and R. Dempsey, *What Works for Women at Work: Four Patterns Working Women Need to Know*. NYU Press, 2014.
- [32] A. Lee and J. C. Carver, "FLOSS participants' perceptions about gender and inclusiveness: A survey," *Proceedings of the 41st International Conference on Software Engineering (ICSE '19)*, pp. 677-687, 2019.
- [33] M. Coccia and I. Benati, "Comparative studies," in *Global Encyclopedia of Public Administration, Public Policy, and Governance*, A. Farazmand, Ed. Springer, Cham, 2018.
- [34] Z. Shen and M. Spruit, "A systematic review of open source clinical software on GitHub for improving software reuse in smart healthcare," *Applied Sciences*, vol. 9, no. 1, 2019.
- [35] I. Hwang, M. Kim, and H. J. Ahn, "Data pipeline for generation and recommendation of the IoT rules based on open text data," in *Proceedings of the 30th International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, 2016, pp. 238-242.
- [36] W. K. Härdle and L. Borke, "GitHub API based QuantNet mining infrastructure in R," in *SFB 649 Discussion Papers SFB649DP2017-008, Sonderforschungsbereich 649*, Humboldt University, Berlin, Germany, 2017.
- [37] Y. Tang, H. Zhou, X. Luo, T. Chen, H. Wang, Z. Xu, and Y. Cai, "Xdeblat: Towards automated feature-oriented app debloating," *IEEE Transactions on Software Engineering*, pp. 1-19, 2021.
- [38] S. Kochanthara, Y. Dajsuren, L. Cleophas, and M. van den Brand, "Painting the landscape of automotive software in GitHub," *MSR 2022*, 2022.
- [39] F. Zanartu, C. Treude, B. Cartaxo, H. S. Borges, P. Moura, M. Wagner, and G. Pinto, "Automatically categorising github repositories by application domain," 2022. [Online]. Available: <https://arxiv.org/abs/2208.00269>
- [40] H. Borges, A. Hora, and M. T. Valente, "Predicting the popularity of github repositories," in *Proceedings of the The 12th International Conference on Predictive Models and Data Analytics in Software Engineering*, 2016.
- [41] P. Greasley, *Quantitative Data Analysis with SPSS*, 1st ed. Milton Keynes, UK: Open University Press, 2008.
- [42] C. Marshall and G. B. Rossman, *Designing Qualitative Research*, 3rd ed. London: Sage Publications, 1999.
- [43] Government of Ontario, "Inclusive Design Toolkit," <https://github.com/ongov/Inclusive-Design-Toolkit>, 2022. [Online; accessed 25-Jul-2022].
- [44] J. Doughman, W. Khreich, M. El Gharib, M. Wiss, and Z. Berjawi, "Gender bias in text: Origin, taxonomy, and implications," in *Proceedings of the 3rd Workshop on Gender Bias in Natural Language Processing*, 2021, pp. 34-44.

- [45] K. Mendes, J. Ringrose, and J. Kellar, “#metoo and the promise and pitfalls of challenging rape culture through digital feminist activism,” *European Journal of Women’s Studies*, vol. 25, no. 2, pp. 236–246, 2018.
- [46] A. Datla, “Leading with empathy: Tarana burke and the making of the me too movement,” *Harvard Kennedy School Case Program*, 2020.
- [47] S. Ayoubi, “FaceLib,” <https://github.com/sajjjadayobi/FaceLib>, 2022, [Online; accessed 15-Jul-2022].
- [48] D. Betebenner, “randomNames,” <https://github.com/CenterForAssessment/randomNames>, 2022, [Online; accessed 25-Jul-2022].
- [49] Z. Yang, “Speaker Gender Classification,” [https://github.com/zhihanyang2022/gender\\_audio\\_classification](https://github.com/zhihanyang2022/gender_audio_classification), 2022, [Online; accessed 25-Jul-2022].
- [50] C. Kronk, “Gender, Sex, and Sexual Orientation,” <https://github.com/Superraptor/GSSO>, 2022, [Online; accessed 25-Jul-2022].
- [51] S. Gergely and P. Peter, “gender\_prediction,” [https://github.com/gszakal/gender\\_prediction](https://github.com/gszakal/gender_prediction), 2022, [Online; accessed 25-Jul-2022].
- [52] S. Lowe, “UK Panel Show Gender Breakdown,” <https://github.com/slowe/panelshows>, 2022, [Online; accessed 25-Jul-2022].
- [53] PyLadies, “PyLadies,” <https://github.com/pyladies/pyladies>, 2022, [Online; accessed 15-Jul-2022].
- [54] GameHer, “GameHer,” <https://github.com/Noelierrx/GameHer>, 2022, [Online; accessed 25-Jul-2022].
- [55] B. V. Campenhout, “maizeUG,” <https://github.com/bjvca/maizeUG>, 2022, [Online; accessed 15-Jul-2022].
- [56] B. Van Campenhout, D. J. Spielman, and E. Lecoutere, “The role of gender in ICT-mediated agricultural information campaigns,” *International Association of Agricultural Economists*, 2018.
- [57] H. B. Belgacem, “French Dictionary,” <https://github.com/hbenbel/French-Dictionary>, 2022, [Online; accessed 15-Jul-2022].
- [58] H. Amin, “CitiBike-Analytics-JerseyCity-2020,” <https://github.com/helamin/CitiBike-Analytics-JerseyCity-2020>, 2022, [Online; accessed 25-Jul-2022].
- [59] Newism Pty Ltd, “craft3-fields,” <https://github.com/newism/craft3-fields>, 2022, [Online; accessed 25-Jul-2022].
- [60] M. Marr, “gender-bias,” <https://github.com/gender-bias/gender-bias>, 2022, [Online; accessed 15-Jul-2022].
- [61] D. Laniado, Y. Volkovich, K. Kappler, and A. Kaltenbrunner, “Gender homophily in online dyadic and triadic relationships,” *EPJ Data Sci.*, vol. 5, no. 19, 2016.
- [62] J. Zhao, “Gender Bias Papers,” <https://github.com/DgCtRbt/GenderBiasPapers>, 2022, [Online; accessed 15-Jul-2022].
- [63] M. Bartl, “Gender bias in BERT,” <https://github.com/marionbartl/gender-bias-BERT>, 2022, [Online; accessed 18-Jul-2022].
- [64] D. Saunders, “gender-debias,” <https://github.com/DCSaunders/gender-debias>, 2022, [Online; accessed 18-Jul-2022].
- [65] H. Liu, “Debiased-Chat,” <https://github.com/zgahhlhc/Debiased-Chat>, 2022, [Online; accessed 18-Jul-2022].
- [66] T. Wang, J. Zhao, M. Yatskar, K.-W. Chang, and V. Ordóñez, “Double-hard debias,” <https://github.com/uvavision/Double-Hard-Debias>, 2022, [Online; accessed 18-Jul-2022].
- [67] T. Wang, J. Zhao, M. Yatskar, K.-W. Chang, and V. Ordóñez, “Balanced-datasets-are-not-enough,” <https://github.com/uvavision/Balanced-Datasets-Are-Not-Enough>, 2022, [Online; accessed 18-Jul-2022].
- [68] A. Bhowmick, “paper-judicial-bias-india,” <https://github.com/devdataalb/paper-justice>, 2022, [Online; accessed 18-Jul-2022].
- [69] E. Ash, S. Asher, A. Bhowmick, D. L. Chen, T. Devi, C. Goessmann, Paul, Novosad, and B. Siddiqi, “Measuring gender and religious bias in the indian judiciary,” *Center for Law & Economics Working Paper Series*, 2021.
- [70] T. M. Timmers, T. M. Willemsen, and K. G. Tijdens, “Gender diversity policies in universities: A multi-perspective framework of policy measures,” in *Higher Education*, vol. 59, 2010, pp. 719–735.
- [71] C. Isaac, B. Lee, and M. Carnes, “Interventions that affect gender bias in hiring: A systematic review,” in *Academic Medicine*, vol. 84, no. 10, 2019, pp. 1440–1446.
- [72] M. Bartl, M. Nissim, and A. Gatt, “Unmasking contextual stereotypes: Measuring and mitigating BERT’s gender bias,” in *Proceedings of the Second Workshop on Gender Bias in Natural Language Processing*, M. R. Costa-jussà, C. Hardmeier, K. Webster, and W. Radford, Eds., 2020.
- [73] R. Van Noorden, “The ethical questions that haunt facial-recognition research,” November 2020, [Online; posted 18-November-2020]. [Online]. Available: <https://www.nature.com/articles/d41586-020-03187-3>
- [74] R. Kumar, A. K. Ojha, S. Malmasi, and M. Zampieri, “Benchmarking aggression identification in social media,” in *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, 2018, pp. 1–11.
- [75] A. Bhargava, “Gender bias in computer software programs: A checklist for teachers,” in *Information Technology in Childhood Education Annual*, vol. 1. Association for the Advancement of Computing in Education (AACE), 2002, pp. 205–218.
- [76] D. Silverman, *Doing Qualitative Research*, 3rd ed., Newbury Park, London, 2009.