2023 IEEE/ACM 16th International Conference on Cooperative and Human Aspects of Software Engineering (CHASE) | 979-8-3503-3734-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/CHASE58964.2023.00015

Understanding information diffusion about open-source projects on Twitter, HackerNews, and Reddit

Hongbo Fang
Carnegie Mellon University
Pittsburgh, USA
hongbofa@andrew.cmu.edu

Bogdan Vasilescu
Carnegie Mellon University
Pittsburgh, USA
bogdanv@cs.cmu.edu

James Herbsleb
Carnegie Mellon University
Pittsburgh, USA
jdh@cs.cmu.edu

***Abstract*—The diffusion of information about open-source projects is a key factor influencing the adoption of projects and the allocation of developer efforts. Developers learn about new projects, and evaluate their quality and importance by accessing the related information. Social media is an important channel for information diffusion about open-source projects, with previous research suggesting the existence of a social media ecosystem that consists of multiple platforms and collectively supports information diffusion in open source.**

With different features supporting information diffusion, the same piece of information likely reaches different developer communities on different platforms, which attracts the attention and contribution of different developers and thus influences the success of open-source projects. Despite its importance, few works looked at the identity of the developer community that project-related information reaches on social media platforms and its associated impact on the discussed project.

In this work, we track social media discussions on open-source projects on three different platforms: Twitter, HackerNews, and Reddit. We first describe the dynamics of project-related information diffusion across platforms, and we analyze the association between the number of posts on each platform, and the number of developers attracted to the discussed project from different communities. We find that posts about open-source projects first appear on Twitter and HackerNews, then move more towards Reddit. The number of project-related posts on Twitter mostly associate with the attracted developers from communities that are close to the project's main contributor, while posts on other platforms associate more with the attention from remote communities.

***Index Terms*—information diffusion, social media, open-source software**

I. Introduction

The development of open-source software is a collective community effort. Unlike industrial software development, there is little, if at all, centralized allocation of efforts and tasks [1]. Often, developers self-organize into working groups [2]. They are free to choose the projects and tasks to work on [3], and decide to join [4] or disengage [5] at their own will. Underlying this allocation of efforts in the open-source community, information plays an essential role. In our context, information is defined as the message that is informative of open-source development activities [6]. We focus on the information about specific open-source projects in our study, as they are more relevant to the attention and contributions at the project level, compared to general discussions about open source as a whole. The diffusion of project-related information makes the project aware by the other developers, which is the basis for future contributions or adoptions [4]. A general introduction of the project draws the attention of the developer community who are in need of such projects, and the information about specific tasks points the developer community to the contributions that the project needs [7]. Information about projects serves as signals to indicate the project quality [8], impact [9], and the activeness of the development and user community [9], [10], all of which influence the developers' adoption and contribution activities. Therefore, the diffusion of project information is important to the success of open-source projects and developers, and the health of the open-source ecosystem overall.

Information about open-source projects can be diffused through multiple channels. For example, the collaboration between developers creates opportunities for information exchange [11], and the structure of developers' collaboration network in open-source ecosystems influences the flow of information, which further affects the onboarding of new developers to projects and the quality of projects developed [11]–[13]. However, as pointed out by Ducheneaut, the success of open-source projects often relies on the support of a large community [14], and the size of the supporting communities can be as large as thousands of developers, if not more [15]. Not all members of this community have close collaboration ties with the projects' core developers [16], thus there should exist other channels of information diffusion that go beyond the close circles of developers reached through direct collaboration experience.

Recently, information diffusion on social media, which is broadly defined as the media channels which support socially-enabled many-to-many communication [17], attracts much attention. The information on social media is usually visible to a large audience, and many platforms provide features for the easy sharing of content (e.g., retweets on Twitter). Those characteristics make social media an ideal place to diffuse and access project-related information. Indeed, researchers

2574-1837/23/$31.00 ©2023 IEEE
DOI 10.1109/CHASE58964.2023.00015

found project-related information on over a dozen social media platforms [18], with a majority of developers posting and consuming project information on social media [19], [20], and almost all popular projects are discussed in the social media space [21].

Information is organized and diffused differently on different platforms. As an example, posts on Reddit are grouped into subreddits based on the themes, and the visibility of content is largely dependent on community signals like votes. [1] In contrast, the content of different topics is usually collapsed on Twitter and the information is mostly diffused through the users' follower networks. [2] Those differences influence the visibility of the posts on different platforms, and the information accessibility of different developers. With previous works empirically evaluating the social media's effect to attract community attention and contributions [21], [22], there have been few works looking into the varied influence different platforms have on the project because of their distinct information diffusion mechanisms.

We argue this is an important gap in the literature. The identity of developers who the information reaches, and who are later likely attracted to the project matters to the project's success. Socially-close developers are more likely to join as a contributor upon receiving the information because of the social closeness [23], and they tend to coordinate better with the existing project team after joining because of the familiarity of work norms [12], [24]. On the other hand, contributors from remote communities are more likely to have different skill sets and knowledge, which increases the diversity in the project team and is critical for team innovation [25]. Understanding the difference in the communities that the information on each media platform reaches contributes to the theoretical understanding of the value of different social media to open-source software, and provides practical suggestions about how to diffuse or access project information on social media based on different needs.

In this work, we track project-related discussions on Twitter, HackerNews, and Reddit since the time of project creation. We first provide an overview of the project information diffusion on each platform, and then conduct regression analysis to understand the association between the number of social media posts on each platform and the number of attracted developers from different communities. We report that posts about the same project tend to appear first on Twitter and HackerNews, then move more towards Reddit. The number of Twitter posts is more associated with the attracted developers from communities that are socially close and technically similar to the project's main contributor; in contrast, the number of posts from HackerNews and Reddit is more associated with the attracted developers from remote communities.

[1]https://reddit.zendesk.com/hc/en-us/articles/7419626610708-How-does-voting-work-on-Reddit-

[2]https://help.twitter.com/en/using-twitter/twitter-timeline

II. Related Works

Early research described the development of open-source software projects as a voluntarily collaborative work. Lacking traditional coordination and work allocation mechanisms, open-source projects are built by developers who self-organize themselves into teams and voluntarily choose works that they consider interesting or important [26], [27]. The exploration and evaluation of projects, and the identification of specific tasks to work on all rely on information about the projects. Previous research identifies the awareness of the project as the first stage of new contributors joining the project [4]. The information about project popularity [8], development stage [28], functions and applications [7], [20] serves as signals for developers to evaluate the quality and importance of the project, and enable them to find the project they want to contribute. In sum, the information flow underlies the attention and effort allocation in open-source software, and is essential to the success of open-source projects.

In open-source communities, information is diffused through multiple channels. Early works by Hahn, Casalnuovo, and others found that developers are more likely to join projects if they shared collaboration experience with the existing developers before [12], [23], [29], and the structure of collaboration networks in the open source community influences the accessibility of information and further affects the success of projects [13]. Peng later explicitly described the collaboration experience as a channel for information flow and empirically showed that it had a stronger influence on project success than project-watching, which is another mechanism to diffuse project-related information [11].

Social media has long been recognized as the platform for information diffusion. Researchers found information about open-source projects from over a dozen social media platforms [17], [18], and the use of social media to diffuse, and access information is ubiquitous both at the developer and the project level [19]–[21]. Similar to the information exchange through collaboration networks, information on social media also influences the success of the projects, with works by Fang et al.causally showing that social media posts help to attract more stargazers and contributors to the mentioned projects [22].

Previous works largely ignore the varied information diffusion process across different platforms, which may lead to different impacts on the mentioned project. Our work addresses this gap by uncovering the different communities that posts on each platform reach, and discussing the varied impact it has on the project mentioned in the social media space.

III. Research Questions

In this work, we first describe how project information is diffused on different social media platforms since the time of project creation, and we explore the difference in developer communities that posts on different social media reach.

To begin with, we ask about the amount of social media discussions on different platforms related to open-source projects.

This volume provides an overview of the extent to which each platform is used for open-source project discussion, and is a straightforward measurement of the importance of different platforms to open-source software development. Therefore, we ask:

RQ1. How much project-related social media discussion is there on different platforms?

Next, the time when posts about a project appear on different platforms is important. Many developers use social media to learn about emerging projects in the community or stay up to date about the latest project news. For those developers, understanding the promptness of information on each platform helps them to better allocate time across platforms. Similarly, the timeliness of information is important for project bug identification and triaging using social media posts. Moreover, understanding the difference in time when posts about the same project appear on different platforms reveals the discussion evolution across platforms, which contributes to the theoretical understanding of how different platforms in the social media ecosystem work together to support project information diffusion. Therefore, we ask:

RQ2. When do social media posts on the same project appear on different platforms?

For open-source projects, a key benefit of being discussed on social media is the attraction of community attention and new contributors. With previous research predominately focused on the amount of attention, little is known about the identity of developers whose attention is attracted. As discussed in section I, the identity of developers attracted matters because of the different value they bring to open-source projects. Therefore, understanding the varied developers different platforms attract provides important implications to project promoters on their choices of promotion platforms, and we ask:

RQ3. Who are the developers attracted to the project by different social media?

IV. Methods

We describe the steps of data collection and analysis below, the data and code to reproduce our results are available online DOI 10.5281/zenodo.7726304 [30].

A. Choice of ecosystems and platforms

In this project, we focus on the social media posts from Twitter, HackerNews, and Reddit that mention open-source projects in R and Python languages. The three platforms selected are popular social media channels adopted by open-source developers for information diffusion and project-related discussions [20], [31], and we select R and Python ecosystems because they are two popular programming languages for open-source development [32], [33], with interesting differences that the Python language produces projects of a wide range of applications while the R language is more specifically used for statistical computing and analysis purposes. [3] [4] Therefore, the two communities likely involve developers, users, and stakeholders of different identities and backgrounds, which leads to different social media usage. The selection of the study subjects follows the guidance provided by Seawright and Gerring that the subject is both representative of the general social media use in open-source, and provides interesting variance that aligns with the research question [34].

B. Data collection

We first collect all projects primarily written in R or Python, and that are not forked projects themselves from GHTorrent dataset [35]. Due to the data hole in the second half of 2019 [5], we restrict our sample to only projects created before 2019-06-01 so that we can get a complete list of projects in the two ecosystems. Following the suggestion in [36], we remove all projects with less than ten commits before the end of the study period because they are likely abandoned early before being completed. After this stage, we identify 63,721 projects written mostly in R, and 688,623 projects written mostly in Python.

Next, to focus our study on open-source software, we use GitHub API to collect the license information of the project. We remove all projects with no license, or license returned as ”others” from the API (because we are not able to validate it as an open-source license), or projects that have been deleted on the platform (because we cannot obtain their license information). There are 8,420 R projects and 177,639 Python projects left after this step.

Finally, we use Twitter academic API [6], Algolia HN search API [7], and Pushshift Reddit API [8] to query the social media posts about the set of projects under study. Following [22], we use *”github.com/repo_slug”* (e.g., *”github.com/torvalds/linux”*) as the query search key (see section VII for a discussion on the selection of the search query) and collect all posts containing this keyword before 2019-06-01.

C. Analyzing the amount of project-related discussion on different platforms

To answer **RQ1**, we compute the total number of posts that mention any one of the R, or Python projects on each platform. For simplicity, the main posts, comments, replies, and retweets are considered separate posts and added up together. However, this simple measurement does not equate to the popularity of each platform among open-source communities because posts can be of varied lengths across platforms. To address this, we also compute the number of unique users who post about projects on each platform, and the number of projects that are discussed on each platform. The result is grouped by years to study the longitudinal change in platform usage across time.

[3] https://www.r-project.org/
[4] https://www.python.org/about/apps/
[5] https://twitter.com/ghtorrent/status/1284402052739878913
[6] https://developer.twitter.com/en/products/twitter-api/academic-research
[7] https://hn.algolia.com/api
[8] https://github.com/pushshift/api

To better understand the importance of social media to the entire open-source ecosystem, in each year, we compute the percentage of projects being discussed on each media platform relative to the total projects created with the following equation:

$$R_{itp} = \frac{N_{itp}^{\text{mention}}}{N_{it}^{\text{create}}}$$

where R_{its} stands for the ratio of projects being discussed in ecosystem i (i = R or Python), in year t, and on platform p, N_{itp}^{mention} is the number of projects in ecosystem i, created in year t, and mentioned on platform p within one-year after project creation (The "within one-year limitation" is added to ensure a fair comparison between projects created at different times because of their different lengths of observation in the study), and N_{it}^{create} is defined as all projects in ecosystem i created in year t.

Note the projects in this analysis need to have at least one year of history, and we only report the longitudinal trend until 2018 because of missing data in later years (The ratio in 2018 was computed only on projects created before 2018-06-01).

D. Analyzing the post appearance time for the same project on different platforms

To answer **RQ2**, For each project, we obtain all social media posts that mention it, and we compute the time when the first post about the project appears on each platform, relative to the project creation time.

Next, we analyze the change in social media usage over time, if any, as the project gets older and becomes not new to the community. For all projects with at least a total of four social media posts from all platforms, we order all posts based on their posting time and split them into four quantiles, with the first quantile being the first 25% posts that appear on any social media platform. We compute the likelihood that posts in each quantile appear on different platforms, which provides insight into the shift in social media usage over time.

E. Analyzing the association between social media posts and the developers attracted to the project from different communities

We operationalize the identity of developers attracted based on their distance from the project's main contributor in the collaboration graph. For each project, we construct a collaboration graph at the time of project creation. A node in the graph represents one developer, and an edge between two nodes indicates that those developers have committed to the same project(s) in the past year. The network distance between two developers in the graph represents both the technical similarities, as developers who work on the same project tend to have similar technical skills; and the social familiarities, as the two developers shared collaboration experience in the past. As discussed in **RQ3**, the technical similarity and social closeness of attracted developers are important to project success as they affect the quality of coordination and the ability of team innovation. Thus understanding social media's influence to attract developers in different network distances provides important insights.

We conduct a regression analysis to identify the associations between the amount of project-related posts on each media platform, and the number of attracted developers in different distance groups. Specifically, We divide all the stargazers (i.e., the developers who starred the project) and the contributors (i.e., the developers who made at least one commit to the project, excluding the top contributor) a project receives before the end of the observation period (i.e., 2019-06-01) into groups based on their network distances to the project's main contributor. The main contributor is defined as the developer who contributes most commits to the project within the first year after project creation, and is often the developer who initiates the project and does most of the project promotion online [7], thus they are not only the key stakeholder of the project, but also a main source of information.

We consider the stargazers and contributors to be attracted to the project in some way, and we conduct regression analysis with the number of project's stargazers (or contributors) in different network distances (to the project's main contributor) before the end of the observation period as the outcome variable, and the number of social media posts on different platforms the project receives in the same period as independent variables. In the model, we also control for the total number of developers in a given network distance, together with other variables. The full list of variables used in our model is shown in table I. The relimp package [37] is used to measure the percentage of variance explained by each independent variable, and the larger the variance explained, the stronger the association between the independent variable and the outcome, controlling for other variables [38]. The main contributors of less than 4% of all projects are not identifiable because the author information of commits is not recorded in the dataset, and those projects are thus excluded in the regression analysis. Overall, 8,189 R projects and 172,915 Python projects are used for this analysis.

Finally, because of the correlational nature of our analysis, we are not able to make causal claims with our results and a two-way effect between the independent variables (social media posts) and the outcome variable (attracted developers) may exist. Specifically, a high association between posts on one social media platform and the number of stargazers (or contributors) in a given distance group may indicate the posts on that platform is better at attracting developers from that group, alternatively that it can be explained by the attracted developers from that group more likely to post on the given media platform. While we are not able to eliminate the effect from the latter explanation with the current research design, we suggest that the alternative effect, if exists, is also an important observation, as it suggests that different social media are preferable to post about the same project by different developer communities. We call upon future causal studies to clarify the confusion and look into the interesting patterns in both directions.

TABLE I: The definitions of the variables used in the model

Outcome variables	
X-hop contributor	The total number of contributors to the project before the end of the observation period who are at X-hop away from the top contributor in the collaboration network. All contributors are grouped into one-hop, two-hop, three-hop, and four-or-more-hop contributors in the study.
X-hop stargazer	Analogous to *X-hop contributor*. Stargazers are defined as developers who star the project.
Social media variables	
Twitter post	The total number of posts mentioning a project on Twitter before the end of the observation period.
Reddit post	The total number of posts mentioning a project on Reddit before the end of the observation period.
HackerNews post	The total number of posts mentioning a project on HackerNews before the end of the observation period.
Control variables	
X-hop developer	Analogous to *X-hop contributor*, we group all developers (or all accounts on GitHub) created before the time of project creation into the four groups, and those developers do not necessarily interact with the project.
Project age	The number of days since the project creation.

F. Qualitative evaluation over social media posts on open-source projects

To provide a richer understanding of the project information flow in the social media space, we conduct a qualitative case-study evaluation of the posts of several open-source projects that are heavily discussed in social media. The first author manually evaluates all the captured posts mentioning the focal project and provides a description of the social media discussion evolvement over time. In addition to the query key used in the data collection step(i.e., *"github.com/repo_slug"*), we also use the project URL (if available) on the project homepage as a second query key (e.g., *mjskay.github.io/tidybayes* for project *mjskay/tidybayes*), and we manually evaluate the obtained posts to remove false positives.

V. Results

A. The amount of project-related discussions on different platforms

Figure 1 reports the amount of project-related discussion on different social media platforms for R and Python projects. Overall, there are more and more social media posts about projects over years on most channels, but the increase has slowed down in recent years. One exception is the number of posts for the R projects on HackerNews, where we observe a decrease since 2016, and a similar decrease is also observed when measured by the number of projects discussed or users posting on HackerNews.

Twitter has a much higher volume of discussion than the other platforms, with the number of posts being at least ten times higher than that of the others. A similar difference is observed in the number of users posting project-related content, and the number of projects being discussed on platforms. Reddit and HackerNews have a similar amount of posts in the early years. Since 2014, the number of posts, users posting, or projects mentioned on Reddit increases much faster than that on HackerNews, and at the end of the observation period (2018), Reddit is the second most-used social media platform among the three studied to diffuse project-related information, and the result is consistent across different usage measurement.

The social media usage is generally consistent between R and Python ecosystems. However, Twitter is much more used than the other two platforms in the R community, with the number of posts, users, and projects mentioned almost 100 times more than that of the other platforms. In contrast, while Twitter is still the most popular platform in the Python community, Reddit is also widely used. In 2018, there are 3,757 projects discussed, and 7,215 users posting on Reddit, which is 52.7% of the number of projects discussed, and 15.6% of the number of users posting on Twitter in the same year.

Figure 2 reports the percentage of projects being discussed on different social media within one year after creation, relative to the total number of projects created. Because of the difference in the ratio scale, we use the left y-axis to present the ratio on Twitter, and the right one for other platforms.

For R projects, the percentage of projects discussed on HackerNews and Reddit is no more than 2% in most of the years, indicating that only a very small portion of projects will be mentioned. Since 2014, the ratio of mentioned projects on those platforms is relatively stable over years. In contrast, the ratio of R projects discussed on Twitter has been drastically increasing over years. Since 2016, close to 40% of all R open-source projects ever created will be mentioned on Twitter at least once within one year after their creation.

We observe a different trend in the Python community. The longitudinal change in the percentage of mentioned Python projects on Twitter and HackerNews is similar and there is a decrease in the ratio until the end of the observation period. The ratio of mentioned projects on Reddit drops to the lowest point in 2013, then followed by a steady increase. Overall, Python projects are less likely to be discussed on social media compared to R projects. In 2018, 11.9% Python open-source projects created will be discussed on any of the three social media within one year after creation, while 44.4% R projects will be discussed at the same time.

B. The time of post appearance about the same project on different platforms

In figure 3, we report the time when posts first appear on different platforms for R and Python communities. We use bootstrap to compute the average first-appearance time and

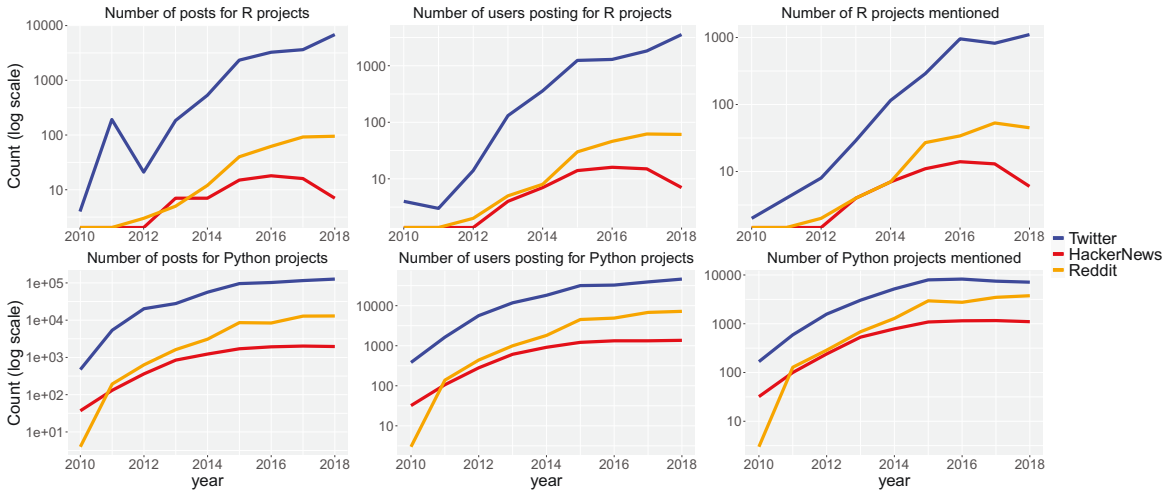


Fig. 1: The amount of project-related discussion on different media platforms across time

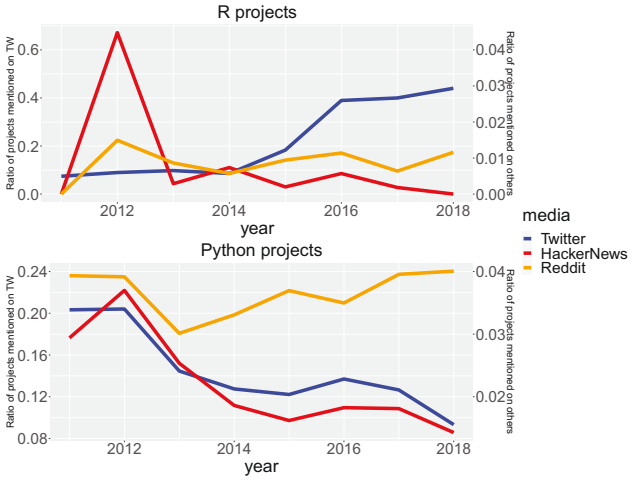


Fig. 2: The ratio of projects being discussed within one year after creation on different social media

the 95% confidence interval of the estimated average time. The red dots represent the estimated mean, and the vertical bars represent the 95% confidence interval.

There is not much qualitative difference between the R and Python ecosystems. On average, posts about a project first appear on Twitter (on average 188 days after creation for R projects and 241 days for Python), then on HackerNews (352 days for R and 377 days for Python), and appear on Reddit the last (508 days for R and 449 days for Python). As a comparison, the first star of a project (excluding the star from the project's main contributor itself) is received on average 286 days after creation for R projects, and 226 days for Python. The time when a tweet first appears is very similar to the time when the project receives its first star, and it suggests Twitter is one of the earliest platforms where developers can learn about a project.

Figure 4 presents the shift in social media use across time. The x-axis indicates the time when a post appears, with the first quantile being the first 25% posts on all media. Similar to section V-A, we use two y-axes of different scales to present the percentage of posts appearing on different platforms. Posts on HackerNews mostly appear among the first quantile of posts discussing the project, or at the very early stage since project creation. Posts on Twitter have a similar pattern to appear more in the early period of the project's lifespan, and are more concentrated in the second quantile in particular. As the project becomes older, the active discussion moves more onto Reddit, indicated by the increase of posts in the third and fourth quantiles. Therefore, we conclude a shift of usage exists that the earliest information about a project first appears on Twitter and HackerNews, and the later-stage information moves toward Reddit. The observed pattern does not vary much between the R and Python ecosystems.

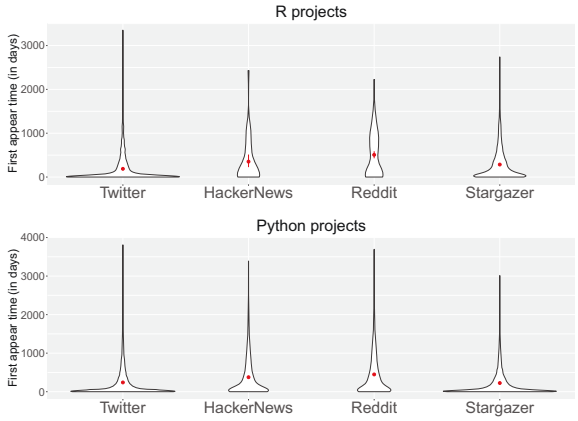


Fig. 3: The time when posts about a project first appear on different platforms, relative to the day of project creation

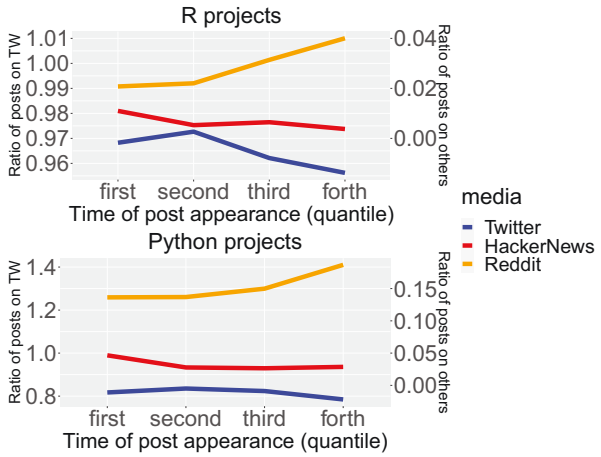


Fig. 4: The shift of media usage across time

C. The correlation between attracted developers in different network distance groups and the number of social media posts

Lastly, we report the association between the number of posts on different social media and the number of stargazers (contributors) attracted to the project, where the stargazers (or contributors) are grouped based on their network distance to the project's main contributor. The estimated coefficients of variables are not presented in the paper because of space limitations and because they are not the main result of interest, but the code and data for generating this result are available in the replication package.

The relative importance of independent variables is summarized in figure 5. Each barplot represents the variance explained by the number of social media posts on one platform (listed in the x-axis), with the outcome variable being the attracted stargazers (or contributors) who are X-hop away from the project's main contributor, and the value of X indicated by the color of the bar. The explained variance is estimated with a bootstrap over possible shuffles of the independent variables and a 95% confidence interval on the average explained variance is computed and presented with the error bar. Note this interval indicates the dispersion of variance explained by the independent variable (or the level of association between the independent variable and the outcome), thus is different from the confidence interval of the estimated coefficient which represents the size of the impact. To interpret the result, a large portion of variance explained by the independent variable corresponds to a high association between this independent variable and the outcome variable, controlling for other variables.

Generally speaking, the more distant the attracted stargazers or contributors are to the project, the higher their association with the number of social media posts. It may suggest that social media's influence goes beyond the close social circles of the project's main contributor, and is generally good at attracting developers from remote communities. Next, the association between the number of social media posts and the number of stargazers the project receives is higher than the association between social media posts and the project contributors. It suggests that the number of social media posts is relatively independent of the attracted developers, which is consistent with previous research that social media has a lower influence to attract new contributors compared with its attraction of new stargazers [22]. We also observe that the association between posts on Twitter and the attracted stargazer, or contributors at any distance is higher than that of other platforms, it aligns with the observation that there are far more posts on Twitter than on the other two platforms, and suggests Twitter likely has the biggest impact to attract new stargazers or contributors, or that more attracted stargazers or contributors will post on Twitter over the other platforms.

One important observation is the relative difference of each social media to explain the number of stargazers (or contributors) attracted from different network distances. For example, in the Python ecosystem, the number of Twitter posts explains a 2.7% variance in the number of stargazers that is one hop away (or directly connected) to the project's main contributor, and the total variance explained by all social media is 3.5%. Therefore, Twitter accounts for 77% of all variance explained by social media posts on one-hop stargazers. In contrast, for stargazers that are four or more hops away, Twitter explains 20.6% variance, with all media explaining a combined 33.9% variance. For those remote stargazers, Twitter only explains 60.7% variance among the all explained by social media.

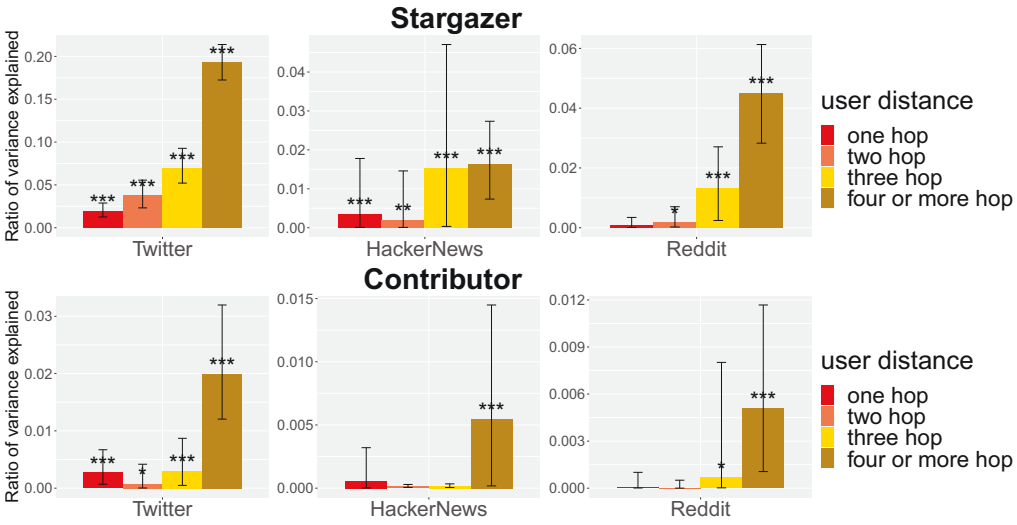
To better illustrate this point, we plot the relative variance explained by each social media, compared to the combined variance explained by all media, for stargazers (and contributors) at different distances in figure 6. Generally speaking, the relative variance explained by Twitter decreases as the distance between attracted stargazers (and contributors) to the main contributor increases. In contrast, Reddit explains relatively more variance for attracted developers from remote communities. On HackerNews, we observe an increase in the relative variance explained in the Python ecosystem as the distance of developers increases, but no major increase in the R community. The observed pattern may be a result of information on different platforms diffusing to different communities, and further attracting different developers to the project.

D. Qualitative analysis on the stream of social media posts

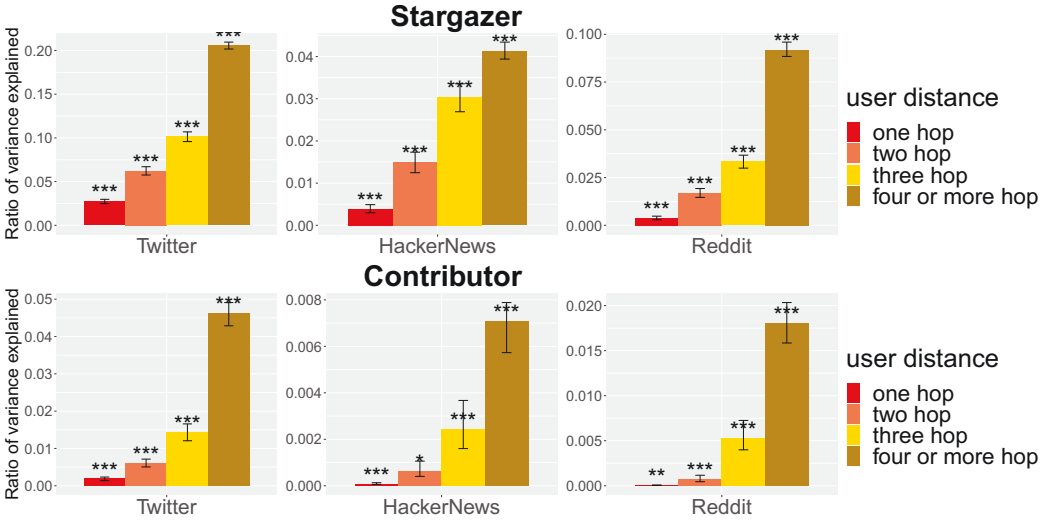
To better understand the open-source information diffusion on social media space, we conduct in-depth qualitative analyses on projects that are heavily mentioned in the social media space. We present the result of one such project, and the collected social media posts for all projects in our sample are publicly released for future research. [9]

The focal project presented is *tidybayes*, which is a popular R project used for Bayesian analysis and visualization. The project received its first commit on 2015-03-29, and obtained

[9]shorturl.at/knHUX



(a) R ecosystem



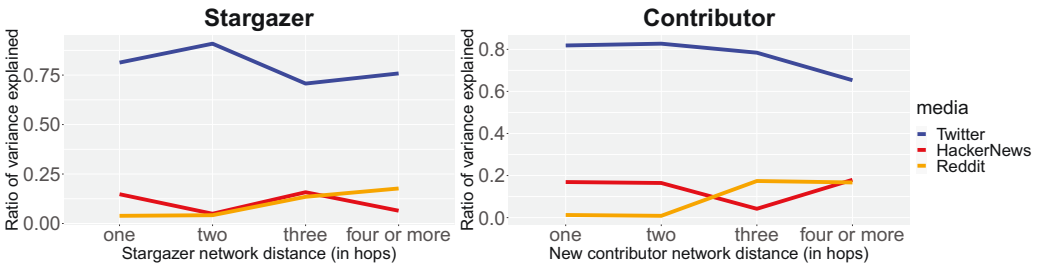
(b) Python ecosystem

Fig. 5: The variance explained by the number of posts on different social media (*p<0.05; **p<0.01; ***p<0.001)

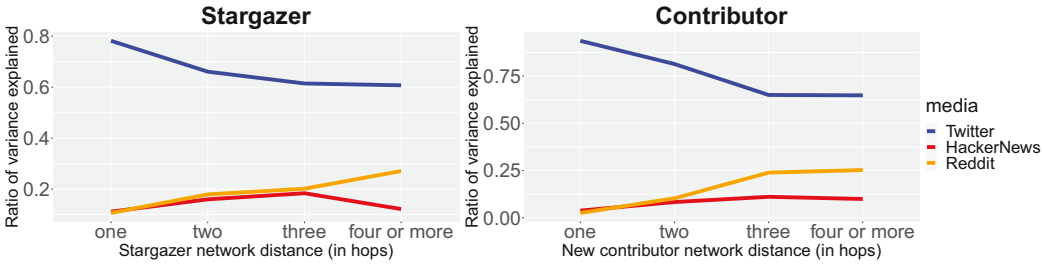
over 300 stars on GitHub at the end of the observation period (2019-06-01). 541 social media posts were captured about the project (four Reddit posts and the rest are tweets), and qualitative analysis reveals three different periods of social media discussion about the project. We describe each period below and provide examples of posts in each period in figure 7.

1) Dormant period: Occasional social media discussion; from 2015-03 to 2017-10: The project was developed by only one developer for the first two and a half years since its creation. During this period, there were only four posts captured in social media (all on Twitter) mentioning the project. The content of the post is mostly general introductions for the project and they are likely posted by the project users.

2) Emerging period: Active project promotion; from 2017-10 to 2018-08: The project gradually moved to a mature and releasable stage within this period and the project owner worked actively to promote it to a larger audience and push it to CRAN package manager. In late 2017, the project owner mentioned on both Twitter and Reddit that he was working on a package to help integrate Bayesian analysis into tidy data analysis. Following this, he introduced the project features in the social media space multiple times. The promotion of the project gained much help from established members in the R community. For example, a member of RStudio tweeted two times to promote the project and those tweets received over 70 retweets in total. In early August, 2018, the project



(a) R ecosystem



(b) Python ecosystem

Fig. 6: The relative variance explained by different social media

owner announced that the project was finally on CRAN, and this message was retweeted 232 times (with another 11 quote tweets) and received 696 likes.

3) Trending period: Project updates and posts from the community; from 2018-08 to 2019-06: In this period, the project is in a fast-development stage where a new version was released about every three months and there were many smaller upgrades. The project owner continued to use social media (mostly Twitter) to inform the community about the recent project updates, and those messages typically received more than ten retweets and was diffused to a large audience. Many project users and open-source developers posted on Twitter to show appreciation for the project or recommend the project to their friends. Other developers also posted on Twitter about the problems they encountered when using the project, where sometimes the project owner, or open-source developers from the community would post a solution.

VI. Discussions

In this paper, we explore project-related information diffusion on Twitter, HackerNews, and Reddit. We identified a different amount of posts, a different temporal pattern of post appearance across platforms, and the likely different communities that information on each platform reach which leads to different developers being attracted to the project. We summarize the main results and discuss the implications below.

A. Social media is widely used to diffuse information about open-source projects

We identify a fairly large number of posts related to open-source projects. The volume of social media posts has been increasing over the years, and recently, there are over ten thousand posts produced on social media about projects in the R or Python ecosystems every year, with at least thousands of unique users participating in the discussion. In both ecosystems, at least 10% of all open-source projects created will be mentioned on one of the three studied social media within one year after the project creation. Note that the social media posts included in our study are even only a subset of all project-related social media posts because we only capture the posts that contain the keywords used as queries, and our estimation of the ratio of projects being discussed is a lower bound of the ground truth. Therefore, We conclude that social media is widely used for project-related discussions, and it plays an important role to diffuse project information.

B. Twitter is the most used platform overall, and there is heterogeneity between ecosystems

There are far more project-related posts on Twitter than the other two media platforms studied, and the result is consistent when measuring the number of users posting, and the number of projects being discussed. However, the usage of media platforms by people from different ecosystems is not the same. We find that Twitter is the predominately used social media among the three studied platforms for R projects, with the number of R project posts on Twitter almost 100 times more than R posts on the other platforms, and over 40% of all open-source projects in R will be discussed on Twitter within one year after project creation. In the Python ecosystem, the discussion of Python projects is more scattered around media channels, with Reddit and HackerNews also used to a certain degree.

Our result is consistent with previous research that Twitter seems to be the most active and widely-used platform for

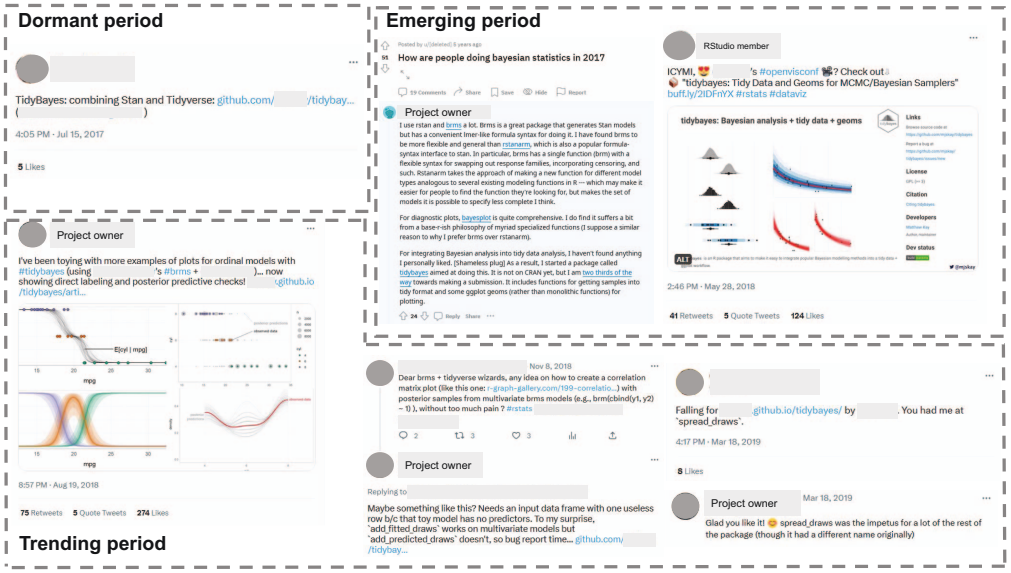


Fig. 7: Tweet samples in each period

project-related discussion [21], but we also suggest that there is heterogeneity between ecosystems and the usage of social media varies for different communities.

C. The timeliness of project information on different platforms vary

Developers may find posts about the same project appear at different times on different platforms. The earliest posts, among the three platforms studied, tend to appear on Twitter, then shortly after on HackerNews. As the project becomes older, more and more project-related information will be found on Reddit.

Many factors may influence the time when posts appear on different platforms. For example, early project promoters may prefer to promote the project on Twitter because the information can be easily diffused to others through their, and their friends' follower networks. However, the visibility of the information on Reddit depends on the community signals like votes, and it may be hard for new projects to get much community attention in the early stage. Community norms may also play a role in the promptness of information posted. HackerNews, as the name suggests, is generally considered a place for news in the open-source community [31].

Regardless of the reason that posts on different platforms appear at different times after project creation, this observation provides important suggestions to open-source practitioners. For open-source developers who use social media to learn about new projects, identify bugs, or for other time-sensitive activities, we suggest that Twitter and HackerNews are better places to get timely information. For developers who want to learn about projects that are created long before, Reddit is also a good platform to seek information from.

D. Social media posts associate more with the number of stargazers attracted, compared to the number of attracted contributors

Previous work by Fang et al.suggests that Twitter posts attract far more stars to the mentioned project than new contributors [22]. In our work, we found a similar pattern on all three social media platforms, that the number of media posts is much more correlated with the number of stargazers of the project, compared to the number of new contributors.

There are two main interpretations of such a result. First, as suggested by the previous study, social media is much better at attracting new stargazers compared to new contributors. It again raises the concern that social media is a double-edged sword for open-source projects, because community attention usually comes along with new requests, and it may overwhelm the existing developers without a proportional amount of new contributors joining. An alternative explanation is that compared to the project's new contributors, most of the posts about projects on social media are authored by the stargazers. This possible explanation, if further validated, contributes to our understanding of the value of passive users to the success of open-source projects, as they collectively do more promotion than the combination of projects' developers.

E. Posts on different social media likely influence different people

With platforms providing different features for information diffusion and the visibility of content depending on different factors, our results suggest they may diffuse information about the same project to different audiences, and thus attract different developers. Specifically, the number of Twitter posts is more associated with the number of stargazers, or new contributors from communities that are close to the project's

main contributor, with HackerNews and Reddit posts more correlated with the attracted developers from remote groups.

We suspect the information diffusion mechanism in each platform may explain the observed pattern. Contents on Twitter are mostly visible to the followers of the post author, and as reported in previous research [7], about 40% of all promotion tweets are posted by the project's owners (high overlap with the main contributor in our study) themselves. Not surprisingly, those tweets will be easily accessible by the developers who are close to the project's main contributors. In HackerNews and Reddit, the visibility of content largely depends on the content popularity in the community through interactions like votes, and the closeness between the information receiver and the author of the post plays a less important role. Therefore, contents are more likely to be diffused to communities that are not necessarily close to the project's main contributor on those two platforms.

This observation provides important implications for open-source project promoters. We suggest they should strategically choose the social media platform to promote based on their specific needs. For example, to promote projects who are in urgent need of new contributors, developers may consider posting on platforms like Twitter where their friends are likely to receive the information and are more willing to help. On the other hand, to find new contributors who have a different skill set from the existing developer teams, a promotion on platforms like HackerNews and Reddit may help more. In addition, both for project promoters and information seekers on social media, we advocate for the use of multiple channels together because it increases the diversity of audiences that the information reaches, and also the diversity of information that developers can gain from social media.

VII. Limitations

There are several limitations to our paper. First, in our data collection, we use all posts that contain given keywords in our analysis, which is a subset of all posts mentioning projects, because posts can discuss projects without explicitly mentioning the selected keywords. We explore several alternative queries and each has its own limitations. For example, we use the slug of the repository as a keyword, but it will return a huge number of false positive results if the project name is a common word in English (e.g., microsoft/Icebreaker, the / is considered a special character and thus ignored in search APIs). Therefore, our reported result is only valid under the set of posts collected, and we suggest future research can validate our results with a different set of query posts, or in a smaller-scale study where the query keyword can be selected on a project-by-project basis and manually remove the false positive results, as what we did in section IV-F.

Second, we use a correlational study to analyze the association between developers attracted from a given network distance, and the number of posts on each social media. This result should not be interpreted as causal and the effect between attracted developers and social media posts can go both ways. We acknowledge it as a limitation of our current study, and suggest future research can clarify the confusion by adopting causal models [22], [39].

Next, we compute the distance between the attracted stargazers (or contributors) to the group and the project's main contributor based on the collaboration graph constructed at the time of project creation, which may differ from the distance computed with the collaboration graph at the time of the star, or new contribution event. While it is almost infeasible to compute the real-time distance at each star or contribution event limited by the computation power, we provide a robustness check by conducting the same regression analysis but only count all attracted developers and social media posts within one year after project creation. The difference between the computed distance (between attracted developers and the project's main contributor) and the ground-truth distance will be smaller in the new model because the star, or new contribution event happened within one year after project creation, and the collaboration network will not change much within a relatively short period. The result is qualitatively similar as reported in the paper, and we include the data and code for the robustness check in the replication package as well.

In addition, we do not consider the difference in the purpose of posts in our analysis, and it is very much possible that the purpose, or content of project-related information is different across platforms. We call upon future research to distinguish the effect caused by the varied purpose of posts and by the different platform information diffusion mechanisms.

There are other limitations to the paper that is common for large-scale empirical social media study. For example, the deleted posts are not available and thus are not included in our sample. We acknowledge the existence of those limitations, but consider it less likely to influence the main result of the paper because of the large scale of our sample and the overall consistency between results in both ecosystems and different measurements.

VIII. Conclusions

To our best knowledge, our paper is the first work to study the open-source information diffusion across platforms, and associate the choice of social media platforms with the identity of developers that the information reaches. Our work unveils the roles that different media platforms play in the diffusion of project-related information by suggesting that social media disseminates information to a broad developer community and goes beyond the close social circle of the project's main developer. Because of the varied information diffusion mechanisms, different platforms may diffuse information to different developer communities, which attracts different developers and influence the success of the project. Our work contributes to the theoretical understanding of the open-source information diffusion process in the social media space, and the social media ecosystem supporting open-source software development. This work also provides practical guidance to open-source developers about how to better use social media to diffuse and access project information.

REFERENCES

[1] A. Bonaccorsi and C. Rossi, "Why open source software can succeed," *Research policy*, vol. 32, no. 7, pp. 1243–1258, 2003.

[2] K. Crowston, "Lessons from volunteering and free/libre open source software development for the future of work," in *Researching the Future in Information Systems*. Springer, 2011, pp. 215–229.

[3] K. R. Lakhani and R. G. Wolf, "Why hackers do what they do: Understanding motivation and effort in free/open source software projects," *Open Source Software Projects (September 2003)*, 2003.

[4] I. Steinmacher, M. A. Gerosa, and D. Redmiles, "Attracting, onboarding, and retaining newcomer developers in open source software projects," in *Workshop on Global Software Development in a CSCW Perspective*, 2014.

[5] C. Miller, D. G. Widder, C. Kästner, and B. Vasilescu, "Why do people give up flossing? a study of contributor disengagement in open source," in *IFIP International Conference on Open Source Systems*. Springer, 2019, pp. 116–129.

[6] A. Madden, "A definition of information," in *Aslib Proceedings*. MCB UP Ltd, 2000.

[7] H. Fang, D. Klug, H. Lamba, J. Herbsleb, and B. Vasilescu, "Need for tweet: How open source developers talk about their github work on twitter," in *Proceedings of the 17th International Conference on Mining Software Repositories*, 2020, pp. 322–326.

[8] H. Borges and M. T. Valente, "What's in a github star? understanding repository starring practices in a social coding platform," *Journal of Systems and Software*, vol. 146, pp. 112–129, 2018.

[9] L. Dabbish, C. Stuart, J. Tsay, and J. Herbsleb, "Social coding in github: transparency and collaboration in an open software repository," in *Proceedings of the ACM 2012 conference on computer supported cooperative work*, 2012, pp. 1277–1286.

[10] H. S. Qiu, Y. L. Li, S. Padala, A. Sarma, and B. Vasilescu, "The signals that potential contributors look for when choosing open-source projects," *Proceedings of the ACM on Human-Computer Interaction*, vol. 3, no. CSCW, pp. 1–29, 2019.

[11] G. Peng, "Co-membership, networks ties, and knowledge flow: An empirical investigation controlling for alternative mechanisms," *Decision Support Systems*, vol. 118, pp. 83–90, 2019.

[12] J. Hahn, J. Y. Moon, and C. Zhang, "Emergence of new project teams from open source software developer networks: Impact of prior collaboration ties," *Information Systems Research*, vol. 19, no. 3, pp. 369–391, 2008.

[13] P. V. Singh, "The small-world effect: The influence of macro-level properties of developer collaboration networks on open-source project success," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 20, no. 2, pp. 1–27, 2010.

[14] N. Ducheneaut, "Socialization in an open source software community: A socio-technical analysis," *Computer Supported Cooperative Work (CSCW)*, vol. 14, no. 4, pp. 323–368, 2005.

[15] K. Crowston and J. Howison, "The social structure of free and open source software development," *First Monday*, 2005.

[16] G. Von Krogh, S. Spaeth, and K. R. Lakhani, "Community, joining, and specialization in open source software innovation: a case study," *Research policy*, vol. 32, no. 7, pp. 1217–1241, 2003.

[17] M.-A. Storey, L. Singer, B. Cleary, F. Figueira Filho, and A. Zagalsky, "The (r) evolution of social media in software engineering," *Future of software engineering proceedings*, pp. 100–116, 2014.

[18] M.-A. Storey, A. Zagalsky, F. Figueira Filho, L. Singer, and D. M. German, "How social and communication channels shape and challenge a participatory culture in software development," *IEEE Transactions on Software Engineering*, vol. 43, no. 2, pp. 185–204, 2016.

[19] S. Black, R. Harrison, and M. Baldwin, "A survey of social media use in software systems development," in *Proceedings of the 1st Workshop on Web 2.0 for Software Engineering*, 2010, pp. 1–5.

[20] L. Singer, F. Figueira Filho, and M.-A. Storey, "Software engineering at the speed of light: how developers stay current using twitter," in *Proceedings of the 36th International Conference on Software Engineering*, 2014, pp. 211–221.

[21] H. S. Borges and M. T. Valente, "How do developers promote open source projects?" *Computer*, vol. 52, no. 8, pp. 27–33, 2019.

[22] H. Fang, B. Vasilescu, H. Lamba, and J. Herbsleb, ""this is damn slick!" estimating the impact of tweets on open source project popularity and new contributors," 2022.

[23] C. Casalnuovo, B. Vasilescu, P. Devanbu, and V. Filkov, "Developer onboarding in github: the role of prior social links and language experience," in *Proceedings of the 2015 10th joint meeting on foundations of software engineering*, 2015, pp. 817–828.

[24] A. Espinosa, R. Kraut, J. Lerch, S. Slaughter, J. Herbsleb, and A. Mockus, "Shared mental models and coordination in large-scale, distributed software development," 2001.

[25] C. R. Østergaard, B. Timmermans, and K. Kristinsson, "Does a different view create something new? the effect of employee diversity on innovation," *Research policy*, vol. 40, no. 3, pp. 500–509, 2011.

[26] A. Mockus, R. T. Fielding, and J. D. Herbsleb, "Two case studies of open source software development: Apache and mozilla," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 11, no. 3, pp. 309–346, 2002.

[27] S. K. Shah, "Motivation, governance, and the viability of hybrid forms in open source software development," *Management science*, vol. 52, no. 7, pp. 1000–1014, 2006.

[28] A. Trockman, S. Zhou, C. Kästner, and B. Vasilescu, "Adding sparkle to social coding: an empirical study of repository badges in the npm ecosystem," in *Proceedings of the 40th international conference on software engineering*, 2018, pp. 511–522.

[29] J. Hahn, J. Y. Moon, and C. Zhang, "Impact of social ties on open source project team formation," in *IFIP international conference on open source systems*. Springer, 2006, pp. 307–317.

[30] H. Fang, , B. Vasilescu, and J. Herbsleb, "Replication package," Mar. 2023. [Online]. Available: https://doi.org/10.5281/zenodo.7726304

[31] M. Aniche, C. Treude, I. Steinmacher, I. Wiese, G. Pinto, M.-A. Storey, and M. A. Gerosa, "How modern news aggregators help development communities shape and share knowledge," in *2018 IEEE/ACM 40th International Conference on Software Engineering (ICSE)*. IEEE, 2018, pp. 499–510.

[32] T. F. Bissyandé, F. Thung, D. Lo, L. Jiang, and L. Réveillere, "Popularity, interoperability, and impact of programming languages in 100,000 open source projects," in *2013 IEEE 37th annual computer software and applications conference*. IEEE, 2013, pp. 303–312.

[33] J. Lai, C. J. Lortie, R. A. Muenchen, J. Yang, and K. Ma, "Evaluating the popularity of r in ecology," *Ecosphere*, vol. 10, no. 1, p. e02567, 2019.

[34] J. Seawright and J. Gerring, "Case selection techniques in case study research: A menu of qualitative and quantitative options," *Political research quarterly*, vol. 61, no. 2, pp. 294–308, 2008.

[35] G. Gousios and D. Spinellis, "Ghtorrent: Github's data from a firehose," in *2012 9th IEEE Working Conference on Mining Software Repositories (MSR)*. IEEE, 2012, pp. 12–21.

[36] 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*, 2014, pp. 92–101.

[37] U. Grömping, "Relative importance for linear regression in r: the package relaimpo," *Journal of statistical software*, vol. 17, pp. 1–27, 2007.

[38] J. A. Rosenthal, *Statistics and data interpretation for social work*. Springer publishing company, 2011.

[39] D. Maldeniya, C. Budak, L. P. Robert Jr, and D. M. Romero, "Herding a deluge of good samaritans: How github projects respond to increased attention," in *Proceedings of The Web Conference 2020*, 2020, pp. 2055–2065.