An Empirical Study of the Apache Voting Process on Open Source Community Governance

Jisheng Wang
Zhejiang University
Hangzhou, China
wangjszz@zju.edu.cn

Lingfeng Bao*
Zhejiang University
Hangzhou, China
lingfengbao@zju.edu.cn

Chao Ni
Zhejiang University
Hangzhou, China
chaoni@zju.edu.cn

ABSTRACT

Open-source software (OSS) projects have become a cornerstone of the software ecosystem, offering numerous benefits to developers and end-users alike. However, ensuring the long-term sustainability and success of OSS projects is challenging, requiring effective community engagement and mentorship. Previous studies have demonstrated that OSS projects benefit from having a larger number of members and an active community, as measured by communication and code contributions. The sustainability of OSS projects must require an effective approach to community governance. In the projects of the Apache Software Foundation Incubator (ASFI), voting plays an important role in community governance, which makes the decision-making process transparent and democratic. Therefore, our study aims to investigate the association between the voting process during the incubation period and the final outcomes of projects (graduated or retired) within ASFI. In this paper, we conduct a comprehensive analysis of the voting process in the ASF projects based on a large-scale data set of the mailing list from 272 sustainability-labeled ASF Incubator projects. We compute various metrics related to voting and investigate whether there are significant differences between graduated and retired projects based on these metrics. We also employ an approach to detect episodic changes based on these voting metrics. Our objective is to examine whether episodic changes in the voting process have a significant impact on the sustainability and success of the project. Our findings reveal that the voting process in the ASF Incubator is closely related to project outcomes, with graduated projects generally exhibiting higher interest and more stability of voting. In practice, these results can help practitioners and project mentors better understand the impact of the voting process on OSS project outcomes and inform strategies to foster a more conducive environment for project success and sustainability within the ASF Incubator and beyond.

CCS CONCEPTS

• **Software and its engineering → Open source model**.

*Corresponding author

Internetware 2023, August 04–06, 2023, Hangzhou, China

ACM ISBN 979-8-4007-0894-7/23/08…$15.00

https://doi.org/10.1145/3609437.3609454

KEYWORDS

OSS Sustainability, Apache Incubator, Mining Software Repository

ACM Reference Format:
Jisheng Wang, Lingfeng Bao, and Chao Ni. 2023. An Empirical Study of the Apache Voting Process on Open Source Community Governance. In *14th Asia-Pacific Symposium on Internetware (Internetware 2023), August 04–06, 2023, Hangzhou, China.* ACM, New York, NY, USA, 11 pages.
https://doi.org/10.1145/3609437.3609454

1 INTRODUCTION

Open-source software (OSS) has become a critical part of software development, making many projects successful through community-driven efforts. The sustainability of an OSS project requires the contribution of many developers and effective governance. Previous experiences show that well-governed communities are more likely to achieve long-term success [10]. To implement effective governance, many OSS projects build standard principles that encourage conversation and community building. However, there are still many OSS projects that fail due to the lack of effective governance mechanisms [8].

To address this challenge, the Apache Software Foundation (ASF) has established the Apache Software Foundation Incubator (ASFI). As one of the most well-known OSS foundations, ASF has incubated a large number of OSS projects by ASFI. ASFI provides a supportive environment for new OSS projects to grow and develop under the guidance of experienced mentors, fostering a strong and vibrant open-source community. It has become an effective way to promote the sustainability of OSS projects and help foster a strong and vibrant open-source community.

Voting is one key means of community governance in the ASFI projects. It is the approach for the members of the ASFI project to reach a consensus, which is a core task for the OSS community. The voting process involves a series of formal votes by the members in the ASFI projects, including mentors, contributors, and members of the ASF's board of directors. These votes are used to make important decisions, such as adding a new feature or determining whether the project is ready to graduate. By voting, the decisions in an OSS project are made in a transparent and democratic manner with all community members. This helps to build trust and collaboration within the community, which is essential for the long-term success of an OSS project.

Some previous studies have studied the sustainability of the ASFI projects by considering some social and technical factors [34, 35]. However, no empirical studies have been conducted to investigate how the voting process affects the governance of ASFI projects. In this study, we collected 56,972 voting-related emails from 272

ASFI projects. Based on the collected data, we explored the importance of the voting process in the sustainability of ASFI projects by answering the following research questions:

RQ1. Are there significant differences in voting measures between graduated projects and retired projects? Likewise, what are the influential voting metrics on the outcome of the projects?

We aim to understand the differences in voting activity between graduated and retired projects. This will help us determine whether a transparent and democratic decision-making process positively impacts the project sustainability. By analyzing the voting metrics and their impact on the outcome of the project, we will gain insights into the effectiveness of community participation and governance mechanisms.

RQ2. What is the impact of voting metrics on other project performance metrics, such as release times and activity within the last few months of the incubation?

Building on the findings of RQ1, the second research question delves into the impact of voting metrics on other project performance metrics, such as release times. By examining the relationships between voting activity and these performance metrics, we can better understand the driving forces behind successful voting activities and how they affect various aspects of project development. We find that various types of voting activity have influential impacts on the defined performance metrics.

RQ3. Are there episodic changes in the voting process of projects during their incubation? How do they differ between sustainable projects and others?

Following the investigation of RQ2, we perform a more comprehensive study on the time series of the voting activity to examine the impact of episodic changes on project sustainability. The third research question focuses on the temporal patterns of voting activity and how they evolve throughout the incubation period. We detect the episodic changes and their change intervals. The findings of RQ3 indicate that episodic changes in voting activities do have a significant impact on project sustainability.

In summary, our paper makes the following contributions:

- We collect a dataset of hundreds of ASFI projects, including their basic information and the corresponding mailing list. All these projects are externally labeled as sustainable or not. We provide a replication package to foster future work, in line with good research practices[1].
- We conduct a deep analysis of voting metrics on the ASFI projects to investigate the differences and significance between graduated and retired projects and identify the important voting metrics related to the project outcome and other performance metrics (e.g., the number of releases).
- We apply the approach of episodic change detection on the voting metrics, and find that sustainable projects adapt better to the changes and handle them more appropriately.

[1]https://figshare.com/s/db3a1c050e7a7ef1fb63

2 BACKGROUND AND THEORIES

2.1 Apache Software Foundation Incubator

As one of the most famous OSS foundations, the Apache Software Foundation (ASF) provides a collaborative and open environment for the development of high-quality OSS software. One of the key initiatives of the ASF is the Apache Software Foundation Incubator (ASFI), serving as a starting point for new OSS projects that want to become part of the Apache community. Under the guidance of experienced mentors, the ASFI offers a supportive environment for young projects to grow. If an ASFI project becomes a top-level project of ASF or a subproject of other existing ASF projects, it will be *graduated*, otherwise *retired*. The ASFI plays a crucial role in building some important principles to ensure the sustainability of new OSS projects, including constructing a strong and diverse community of contributors, maintaining a clear and well-documented codebase, and establishing governance structures that allow for effective decision-making and conflict resolution, etc.

2.2 Apache Voting Process

Voting is an important way of project governance and decision-making in the ASFI projects, and is used to make important decisions in ASFI projects, such as procedural issues, code modifications, releases, and graduation[2]. The voting process in an ASFI project involves a series of formal votes by project members including mentors. Through voting, the project communities can avoid conflicts and reach consensus. The voting process in ASFI projects is usually as follows: First, to reach a consensus across the project community on some important things (e.g., adding features), a vote needs to be initiated. Then, project members cast their votes using a +1 (positive), 0 (neutral), or -1 (negative) system. Finally, the result of the voting is determined based on the majority or a predefined threshold. The voting process is conducted via email.

The voting process is critical to the sustainability of the ASFI projects. First, it ensures transparent and democratic decision-making with community input, fostering trust and collaboration. Second, it ensures projects meet Apache's standards for long-term success, granting access to resources and support upon graduation. Finally, it establishes governance structures, including decision-making and conflict-resolution mechanisms, to manage complex issues in community-driven software development.

2.3 OSS Projects Sustainability

While OSS projects are incredibly successful, they also face unique challenges when it comes to sustainability. Because the code is open and freely available, there is no single owner or company that is responsible for maintaining and supporting the project, as individuals who cannot be easily excluded from using shared natural resources often have little incentive to contribute to the production or maintenance of these resources [24]. Instead, the burden of sustainability falls on the community of contributors who work on the project, often in their spare time and without any financial compensation.

Project sustainability becomes increasingly important in enterprises [26]. Sustainability in OSS projects involves building diverse

[2]https://www.apache.org/foundation/voting

Table 1: Extracted voting metrics

Metric Name	Description
all_total_votes	Total number of votes (both 'yes' and 'no') from all members
all_yes	Total number of 'yes' votes (indicating support) from all members
all_no	Total number of 'no' votes (indicating opposition) from all members
all_vote_times	Total number of voting events from all members
all_vote_results	Total number of emails about the conclusion of the voting result from all members
all_votes_per_month	Average number of votes (both 'yes' and 'no') from all members per month
mentor_total_votes	Total number of votes (both 'yes' and 'no') from mentors
mentor_yes	Total number of 'yes' votes (indicating support) from mentors
mentor_no	Total number of 'no' votes (indicating opposition) from mentors
mentor_vote_times	Total number of voting events from mentors
mentor_vote_results	Total number of emails about the conclusion of the voting result from mentors
mentor_votes_per_month	Average number of votes (both 'yes' and 'no') from mentors per month

communities, maintaining well-documented codebases, and establishing effective governance structures. Success and sustainability are related in many aspects, but they are distinct concepts[36]. OSS success is measured statically, while sustainability is measured dynamically[34]. Sustainability places greater emphasis on social and human factors, such as collaborative decision-making and a welcoming atmosphere, as opposed to technical aspects like code quality or innovation. Research into OSS project sustainability can present actionable insights for project maintenance[27].

3 METHODOLOGY

In this section, we presented our data collection and introduce our method to analyze the data we retrieved to give an answer to the RQs we presented above.

3.1 Data Collection

First, we collect the ASFI project list from Apache Project Podlings[3]. Also, we extract the basic information for each project, including the status (i.e., *graduated*, *current*, and *retired*), the date of entering and exiting (if exists) the incubator, and the mentor list. Then, we implement a web crawler using the Python *selenium* package to automatically download the mailing list for the ASFI projects, which contain different channels for specific purposes (e.g., development, commit submission). Our web crawler can retrieve the emails of a project channel on a monthly basis. Finally, we use *Grimoirelab Perceval*[4], which is an OSS tool for gathering data from software repositories, to store our collected data. The data in our study were collected at October 2022, containing 339 projects in total. Among these projects, 231 projects have graduated, 72 have retired, and 36 are still in incubation. Our study only focuses on the graduated and retired projects.

However, we find that many projects have graduated or retired in a pretty short time due to some external reasons such as becoming a plugin or subproject of another project, so we exclude them from our study. For example, the *Merlin* project only existed in the incubator for only one day and had one release, then it was integrated by the *avalon* project (an early Java Apache Server Framework) and labeled as a graduated project. Besides, we also filter the projects that have very few or no records on the mailing list. An example is *cotton* project which has few mails in the mailing-list, it has just one voting event to purpose retirement because of lack of activity. Finally, 272 ASFI projects remained, including 218 graduated projects and 54 retired projects. Among these projects, there are 1,095,261 email records and 2,912,877 commit records.

Then, we identify the voting-related emails from the collected emails, which cover three different kinds of voting events:

- **A formal call for votes** start a voting event and simply describe the content of the proposal. We used a regular expression to identify the emails whose subject contains the keyword "[VOTE]" but does not starts with "re" (indicating a response to a previous message).
- **Votes cast by project members** are corresponding to the emails that respond to the emails that call for votes, i.e., the emails whose subject starts with "re" and contain "[VOTE]". Because in the ASFI projects, members can only cast a vote by replying to the emails that call for votes. Furthermore, we also identify the mentors' votes according to the mentor list of the ASFI project.
- **Conclusion of the voting result** refers to the email that contain the final result of a round of voting. The subjects of such emails contain the keyword "[RESULT]".

Finally, we collected 56,972 voting-related emails, including 6216 formal calls for votes, 49,278 votes and 1478 voting result conclusions. Among the emails, 51,642 are generated in graduated projects while 5,330 are generated in retired projects. Table 1 shows the list of the voting metrics and the corresponding description.

3.2 Voting Metrics

To investigate the impact of the voting process on the final outcome of Apache incubator projects, we extract several voting-related metrics based on our collected data. Table 1 presents these metrics and their corresponding descriptions.

First, we calculate some statistics of voting events over the entire incubation process of a project. For each project, we count the number of votes cast by project members (*all_total_votes*), which indicates the overall level of the project community involvement in decision-making. Moreover, we count the number of "yes" and "no"

[3]https://incubator.apache.org/projects
[4]https://github.com/chaoss/grimoirelab-perceval

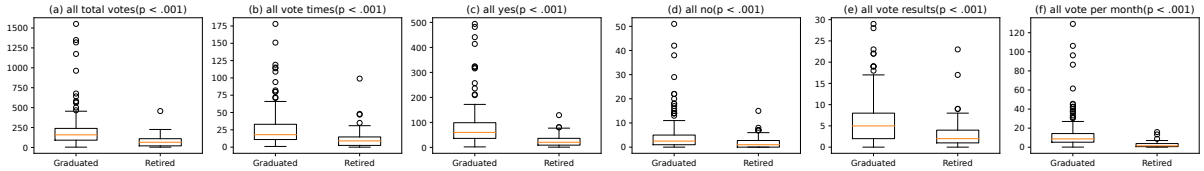

Figure 1: Some descriptive variables between graduated projects (left) and retired projects (right).

votes (*all_yes* and *all_no*), respectively. In the ASFI projects, votes are represented as numbers between -1 and +1, with '-1' meaning 'no' and '+1' meaning 'yes'. We compute the number of voting events (*all_vote_times*) by counting the calls for votes. Similarly, we compute the number of votes that reach an agreement by counting the emails of the conclusion of the voting result (*all_vote_results*). We find that not all voting events have such emails because of some implied emails that disagreed with the vote or intercepted the vote. Some of the voting events automatically terminate without a result conclusion. So it might represent the activity of the whole project to a certain extent. Finally, we compute the average number of votes per month, which might indicate the activeness of project members in participating in the governance of a project.

In the ASFI project, mentors are usually more influential than common project members and their votes might have a greater impact on decision-making. So, we compute the same voting metrics for project mentors, see the lower part of Table 1.

3.3 Data Statistics

In this section, we present some final statistics for our collected data as shown in Figure 1. The metrics used in the plots reflect the voting activity in some extent, and these metrics are part of the metrics that have described before. We can see clearly that graduated projects are more active in voting process. Moreover, the voting activity in graduated projects is more fluctuating, which means that some graduated projects have extremely high voting activity. The selected metrics all have significant influence on the sustainability of the projects (graduated or retired) since its p-value (calculated by the Mann Whitney U test [22]) is less than 0.001. In section 4, we will perform a more complex analysis on these metrics.

3.4 Episodic Changes Detection

In this study, we want to perform a time-series analysis of the voting process of the ASFI projects. Previous studies have shown that many OSS projects experienced large and episodic changes over time [34]. Hence, we follow the study of Yin et al. [35] to identify episodic changes based on some voting-related metrics. Their approach is based on the Cumulative Sum (CUSUM) algorithm [13]. The CUSUM algorithm is a well-established statistical method employed for detecting shifts in the mean or variance of a process or system and has been widely used in various domains. The CUSUM algorithm can be applied to analyze the monthly voting metrics of the Apache Incubator projects, enabling the identification of potential shifts in the voting patterns or project support. In our study, we perform this algorithm on the monthly data of all the voting metrics of all participation except "all_votes_per_month",

Figure 2: Change Intervals detection result.

apart from that, we extract the monthly number of active voters as "active_voters". We define active voters as those who have voted at least once within the last three months. By continuously monitoring a sequence of monthly voting data points, the CUSUM algorithm calculates the cumulative sum of deviations from a pre-determined target value or reference level. When the cumulative sum surpasses a predefined threshold, it might indicate a significant change in the underlying voting process or community sentiment. This real-time detection capability is valuable for applications that require ongoing monitoring and swift responses to changes. Additionally, it uses a drift parameter to penalize a long, flat drift[35]. The threshold and the drift parameter are dynamically selected according to the situation of each project, as referenced in previous research [35]. Figure 2 presents an example of change intervals detected based on the monthly values of "all_yes" in *Abdera* project. In this figure, the upper part shows the time series of the monthly metric value of a project in the incubation time, and the bottom part is the cumulative sums of the positive and the negative change values. When one of the values reaches the threshold, it marks the point as an alarm point, it is also the start point of the episodic change. And the episodic change ends when the specific value gets back to normal. Apparently, the figure represents that the chosen project occurred three episodic changes with regard to the specific metric in the incubation time.

In summary, the CUSUM algorithm provides a robust and effective method for detecting shifts in the mean or variance of monthly

voting metrics for Apache Incubator projects. By continuously monitoring deviations from a target value and comparing the cumulative sum to a predefined threshold, the algorithm can identify potential changes in the voting process, providing valuable insights into the factors influencing project outcomes.

4 RESULT

4.1 RQ1. Differences in voting measures

Motivation. In this section, we aim to analyze the relationship between various metrics and the success and sustainability of open-source software projects in the Apache Incubator. By exploring descriptive statistics, and differences in voting activities, our study seeks to understand the factors that contribute to the graduation or retirement of projects.

Methodology. First, we compare the graduated and retired projects by the descriptive statistics of the voting metrics. Then, we apply the Mann Whitney U test [22] to measure whether the difference in voting metrics between graduated projects and retired projects is statistically significant. We also use Cliff's delta [7][5], which is a non-parametric effect size measure, to show the effect size of the difference between the voting metrics of graduated projects and retired projects.

Besides, we perform a correlation analysis to identify the high-correlated voting metrics and build a regression model to measure the importance of the voting metrics. To ensure a more robust analysis for the regression model, we remove some of the metrics that are high-correlated with the others by the correlation analysis. We apply Spearman's rank correlation test due to its increased robustness when handling data that does not follow a normal distribution [37]. In our analysis, each metric pair displayed a Spearman's rank correlation coefficient consistently below 0.7 [37], indicating a moderate correlation between the variables under investigation. Then, to model the relationship between the metrics and the final outcome of the project (i.e., graduated and retired), we build a logical regression model, which is used in many previous studies [39].

Table 2: Statistics of graduated projects and retired projects.

	graduated		retired	
metric_name	mean±st.dev	median	mean±st.dev	median
incubator_month	21.18 ± 14.07	17.0	39.44 ± 21.49	38.0
all_total_votes	200.0 ± 206.22	156.0	82.0 ± 81.14	65.5
all_vote_times	24.0 ± 24.82	17.0	12.0 ± 16.16	9.0
all_vote_results	5.0 ± 5.57	5.0	3.0 ± 4.12	2.0
all_yes	78.0 ± 76.64	59.0	27.0 ± 25.5	20.5
all_no	4.0 ± 6.48	2.0	1.0 ± 2.83	1.0
all_vote_per_month	12.59 ± 15.8	8.62	2.59 ± 3.08	1.49
mentor_total_votes	29.0 ± 27.18	25.0	19.0 ± 25.61	12.5
mentor_vote_times	1.0 ± 3.61	1.0	2.0 ± 6.63	0.0
mentor_vote_results	0.0 ± 1.0	0.0	0.0 ± 1.41	0.0
mentor_yes	10.0 ± 10.91	9.0	5.0 ± 6.16	4.0
mentor_no	0.0 ± 1.0	0.0	0.0 ± 1.0	0.0
mentor_vote_per_month	1.98 ± 2.68	1.32	0.77 ± 1.79	0.38

Results. Table 2 presents the statistics of the monthly voting metrics in our collected data, including mean±standard deviation and median. In terms of these voting metrics of project members, projects that graduate from the incubator typically have a more active voting process and higher voter participation. For example, members in graduated projects tend to be more involved in decision-making, as evidenced by the higher mean number of +1 votes (78) compared to retired projects (27). Similarly, general voting participation in graduated projects is much more active and has a wider distribution than in retired projects. For the voting actions taken by mentors, we can see something different. Mentors tend to start and end fewer voting events, which means that perhaps there might exist a group of members who are appointed to be responsible to check out the proposals, open a vote and end the vote. This can also be seen from our manual inspection of some projects that most of the vote events are proposed by a group of developers. So it's reasonable for a project's proposals to be made by some core developers in the community and scrutinized by the mentors; this reason makes more sense because apparently the mentors in graduated projects cast more votes than in retired projects.

[5]Cliff defines a delta that is less than 0.147, between 0.147 and 0.33, between 0.33 and 0.474, and above 0.474 as negligible, small, medium, large effect size, respectively.

Table 3: Statistics of graduated projects and retired projects' p-value and corresponding cliff's delta.

metric name	p-value	cliff's delta
incubation_month	<.001	-0.53 (Large)
all_total_votes	<.001	0.56 (Large)
all_vote_times	<.001	0.47 (Medium)
all_vote_results	<.001	0.37 (Medium)
all_yes	<.001	0.63 (Large)
all_no	<.001	0.33 (Small)
all_vote_per_month	<.001	0.78 (Large)
mentor_total_votes	<.001	0.35 (Medium)
mentor_yes	<.001	0.43 (Medium)
mentor_vote_per_month	<.001	0.56 (Large)
mentor_vote_times	>.05	0.04 (Negligible)
mentor_vote_results	>.05	-0.01 (Negligible)
mentor_no	>.05	0.07 (Negligible)

In summary, our data analysis reveals distinct differences in voting statistics between graduated and retired projects. This comprehensive examination lays the groundwork for further exploration into the factors influencing Apache Incubator project outcomes.

Table 3 presents p-values and Cliff's delta values for the voting measures by comparing the ASFI graduated and retired projects. In terms of the majority of voting metrics, the difference between graduated and retired projects is significant except for several voting metrics on mentors (e.g., mentor_vote_times, mentor_no). Furthermore, the effect sizes of the differences for most voting metrics are not negligible. These findings suggest that there are statistically significant differences between graduated and retired projects, indicating that the voting metrics play an important role on the success of ASFI projects.

Figure 3 shows the results of the correlation analysis. We remove four metrics (mentor_yes, mentor_total_votes, all_yes, and all_vote_times) because the values of the metrics are highly correlated with some other metrics. Table 4 shows the regression coefficients of all the voting metrics for the logical regression model

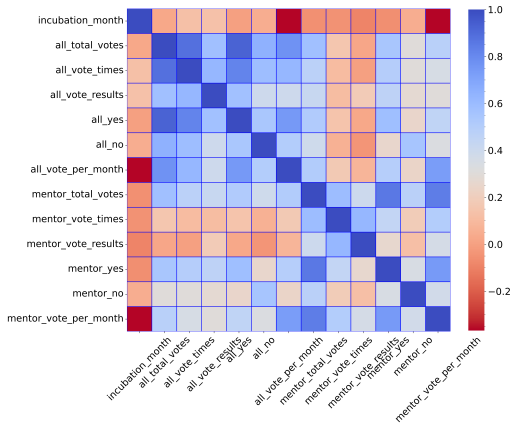

Figure 3: Correlation analysis of the metrics.

built for the outcome of the projects. As shown in the table, the top four metrics that have the highest regression coefficient are mentor_vote_results, mentor_vote_per_month, mentor_no, and all_vote_results.

- **Mentor_vote_results** has the highest positive coefficient indicating that an increase in the number of votes results concluded by mentors positively affects project outcomes. The higher activity of this voting type might indicate that the mentors are actively involved in the project's decision-making and assessment process. This could lead to better project guidance, resulting in a higher likelihood of graduation.
- **Mentor_vote_per_month** has the second-highest magnitude among the coefficients, suggesting a strong influence on project outcomes. A positive coefficient implies that a higher frequency of mentor votes is positively associated with project outcomes. Active mentor engagement may result in better guidance, support, and expertise, which are essential factors for project success.
- **Mentor_no** also has a positive coefficient for this metric, suggesting that an increase in the proportion of "no" votes by mentors is associated with better project outcomes. It could be explained by the fact that mentors who are actively voicing their concerns and identifying issues are also providing valuable feedback that can help the project team address problems and make improvements, ultimately leading to a higher chance of project success.
- **All_vote_results** also has a positive coefficient. Projects with more transparent voting processes and clear communication of results are more likely to graduate. Projects with more transparent vote result emails may encourage active participation from all community members.

Finding. The voting metrics have an influential impact on project outcomes. Many voting metrics have a positive relationship with the graduation of an ASFI project. Projects with active mentor participation in voting events, clear communication of voting results, and transparent decision-making processes are more likely to graduate.

Table 4: Regression coefficients of the voting metrics from the logical regression model

metric name	coefficient	p-value
mentor_vote_results	0.0592	>.05
mentor_vote_per_month	0.0544	<.01
mentor_no	0.0340	>.05
all_vote_results	0.0249	<.001
mentor_vote_times	-0.0201	>.05
all_no	-0.0195	<.05
incubation_month	0.0092	<.001
all_vote_per_month	0.0069	>.05
all_total_votes	0.0008	<.05

Table 5: Descriptive statistics of graduated projects and retired projects on the selected performance metrics.

	graduated		retired	
metric_name	mean ± st.dev	median	mean ± st.dev	median
release_times	2.00 ± 2.83	2.00	1.00 ± 2.45	0.00
last_6_month_commits	1,822.00 ± 3,805.12	605.50	73.00 ± 296.96	5.00
last_6_month_developers	3,293.00 ± 4,162.32	1804.00	116.00 ± 157.20	59.00

4.2 RQ2. Voting metrics' impact on other performance metrics.

Motivation. In RQ1, we identify some voting metrics that have the highest influence on the final outcomes of ASFI projects. In this RQ, we are interested in the relationship between the voting metrics and other performance metrics of the ASFI projects. Because we find that a lot of the retired projects had to vote to retire at the end of the incubation due to a lack of activity in the past few months or they published too few releases.

By examining these relationships, stakeholders can better identify the areas where they need to focus their efforts to improve project outcomes. Additionally, this analysis could help reveal the significance of the voting process in shaping various aspects of project performance and contribute to refining incubation processes for open-source projects.

Methodology. To examine the impact of voting metrics on project performance, we select the following metrics of the ASFI projects:

- **Release_times**: the number of releases published during the incubation period.
- **Last_6_months_commits**: the number of commits extracted from the mailing list during the last six months of the incubation period.
- **Last_6_months_developers**: the number of developers involved in the mailing list during the last six months of the incubation period.

For projects with total incubation months of less than 6 months, the time interval is the whole incubation period. *release_time* is a measure of indicating the project productivity in the incubation period directly, while the other two metrics indicate the activeness and performance of a project at the end of the incubation. We use a similar approach to investigate the relationship between voting metrics and the selected performance metrics.

Table 6: Regression coefficients of the selected performance metrics

Metric Name	Coef.	p-value
mentor_vote_results	-0.0174	>.05
all_vote_results	0.0144	<.001
mentor_vote_per_month	0.0076	>.05
all_no	0.0038	>.05
mentor_vote_times	0.0020	>.05

(a) release number

Metric Name	Coef.	p-value
mentor_no	0.0161	<.001
mentor_vote_results	-0.0119	<.05
mentor_vote_per_month	0.0053	>.05
all_no	-0.0033	<.05
mentor_vote_times	0.0020	>.05

(b) last_6_month_commits

Metric Name	Coef.	p-value
all_no	-0.0067	<.01
all_vote_results	0.0055	<.01
mentor_no	0.0053	>.05
all_vote_per_month	0.0034	<.001
mentor_vote_results	-0.0030	>.05

(c) last_6_month_developers

Results. Table 5 presents the statistics of the performance metrics on graduated and retired projects. As shown in the table, in terms of all three performance metrics, the graduated projects have a higher value than the retired projects. This suggests that successful projects tend to have more active development and collaboration during the incubation period.

Table 6 shows the top 5 regression coefficients of the voting metrics on *release_number*, *last_6_month_commits*, and *last_6_month_developers* from the regression models, respectively. In these tables, we sort the records by the coefficient values. As shown in Table **??**, the relationship between *all_vote_results* and the number of release numbers is positive and statically significant. But the relationship between the other four metrics and the number of release numbers is not significant. This indicates that projects with more voting results during their incubation period tend to have a higher number of releases. This finding makes sense because if a project has more voting events ending with a result email, it will have definitely more proposals passed, resulting in more releases being published.

For the number of commits in the last six months of the incubation period (*last_6_month_commits*), its relationship with *mentor_no* is positive and significant, indicating that projects with more opposing votes from mentors are more likely to have increased activity in the form of commits. This could be because the project team is actively working on addressing the concerns raised by mentors. On the other hand, the relationship between *mentor_vote_results* and *last_6_month_commits* is negative and significant. One possible reason is the project's decision-making process is complex and time-consuming, which slows down the development process and results in fewer commits in the late stage. The relationship between *all_no* and *last_6_month_commits* is also negative and significant, indicating that an increase in the number of negative votes is associated with a decrease in the number of commits. This might be because negative feedback potentially discourages the project teams, leading to a decreasing number of commits.

For the number of developers in the last six months of the incubation period (*last_6_month_developers*), its relationship with *all_no* is negative and statistically significant, indicating that projects with more negative votes tend to have fewer developer interactions engaged in the last six months of the incubation period. Negative votes might be a signal of potential issues or concerns in the project team, making developers have no confidence in the graduation of the project. The relationship between *all_vote_result* and *last_6_month_developers* is positive and significant, which shows that a higher level of engagement in the voting process is associated with the overall performance of a project. For *all_vote_per_month*, the conclusion is the same. However, for the two mentor voting-related metrics, the relationship is not statistically significant.

Table 7: Statistics of monthly episodic changes.

	graduated		retired	
metric_name	mean ± st.dev	median	mean ± st.dev	median
length_yes	3.13 ± 4.3	2.0	8.61 ± 7.0	7.0
count_yes	1.28 ± 1.77	1.0	3.37 ± 2.62	3.0
perc_yes	0.13 ± 0.15	0.08	0.21 ± 0.12	0.22
length_no	2.78 ± 3.78	0.0	3.81 ± 5.03	0.0
count_no	1.11 ± 1.48	0.0	1.49 ± 1.99	0.0
perc_no	0.11 ± 0.13	0.0	0.09 ± 0.1	0.0
length_total_votes	3.02 ± 4.39	1.0	8.33 ± 6.2	7.0
count_total_votes	1.23 ± 1.73	1.0	3.3 ± 2.49	2.0
perc_total_votes	0.11 ± 0.14	0.06	0.21 ± 0.11	0.22
length_vote_times	3.29 ± 4.61	2.0	8.3 ± 6.99	7.0
count_vote_times	1.28 ± 1.8	1.0	3.14 ± 2.72	3.0
perc_vote_times	0.13 ± 0.16	0.07	0.2 ± 0.14	0.2
length_vote_results	2.15 ± 3.67	0.0	5.11 ± 5.26	5.0
count_vote_results	0.87 ± 1.41	0.0	2.04 ± 2.12	2.0
perc_vote_results	0.08 ± 0.12	0.0	0.13 ± 0.12	0.15
length_active_voters	5.52 ± 5.52	4.0	10.14 ± 7.95	8.0
count_active_voters	1.49 ± 1.54	1.0	3.25 ± 2.46	3.0
perc_active_voters	0.25 ± 0.22	0.23	0.25 ± 0.12	0.27

Finding. We find that efficient decision-making processes, increased communication, and active mentor involvement in the voting process positively influence project performance metrics. Negative votes by mentors can be constructive to the progress of the project, but the ones by others tend to be the barrier. These findings indicate that voting metrics play a crucial role in determining the performance of ASFI projects. Combining mentor feedback and overall communication during the voting process can get higher performance.

4.3 RQ3. Episodic voting changes and variations

Motivation. In a previous study, temporal persistence is examined to exist in the relationship between sustainability and some change rates of specific actions [17]. In this RQ, we aim to investigate the characteristics of the voting process based on the monthly voting metrics during their incubation period. So, we follow the approach of Yin et al. [35] to identify episodic changes based on the monthly voting metrics and analyze how these changes may affect the projects' outcomes. As described in Section 3.4, episodic changes are a type of organizational change that occurs infrequently, discontinuously, and intentionally [9]. By detecting the change intervals of each monthly voting metric and performing a time series analysis on voting metrics, we can get a deeper understanding of the

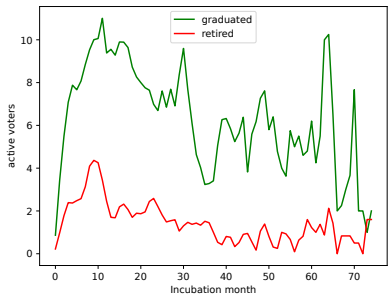

Figure 4: Monthly trend of the number of active voters averaged by all the projects (p-value < .001).

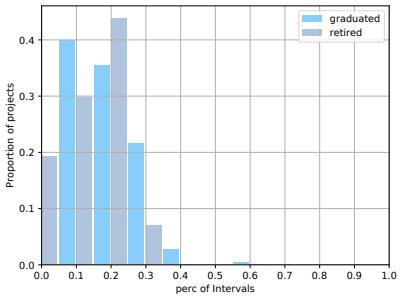

Figure 5: Distribution of the change interval percentage

evolving nature of the voting process throughout the incubation period and its implications on project outcomes.

Methodology. For the voting metrics, including *all_yes*, *all_no*, *all_total_votes*, *all_vote_results*, *all_vote_times*, *active_voters* (see Section 3.2), we calculate a monthly value for each project. Then, we follow the approach of Yin et al. [35] to detect the change intervals (see Section 3.4) and calculate three types of metrics based on the detected change intervals:

- **Length**: The aggregation of length in the month of every change interval on one metric averaged for each project.
- **Count**: The total count of change intervals for each project.
- **Percentage**: The proportion of the change interval "length" to the total incubation time for each project.

For example, a project's incubation time is 24 months and we detect three change intervals on the *all_yes* metric with the corresponding lengths 3, 2, and 4 months for it. So, the "length" metric (denoted as *length_yes*) is equal to 9 months, the "count" metric (denoted as *count_yes*) is equal to 3, and the "percentage" metric (denoted as *perc_yes*) is equal to 37.5% (9/24). We present some statistics of the three types of metrics to figure out the difference between the change intervals in the graduated and retired projects.

Results. Table 7 shows the statistics of the metrics related to change intervals. As shown in the table, we find that graduated projects generally experience shorter and fewer episodic changes (i.e., "length" and "count" metrics). For the percentage of change intervals for the voting metrics, graduated projects also exhibit lower values compared to retired projects. The results on the metrics of change intervals indicate that graduated projects experience a more consistent voting process compared to retired projects. Such consistency may foster a supportive and collaborative environment, enhancing the chances of project success.

Furthermore, we show the results of the time series analysis on the *active_voters* metric, as an example to illustrate the impact of the voting process on ASFI projects. Fig. 4 presents the monthly trend of the averaged number of active voters (*active_voters*) for graduated and retired projects. The number of active voters in the graduated projects is significantly larger than that in retired projects (p-value < 0.001). This result makes sense and is similar to the previous work that finds out graduated projects may be more attractive to the developers[34]. As shown in this figure, the gap in the number of active voters between graduated and retired projects is significant at the beginning of the incubation period. But when the ASFI projects run for a long time, this gap decreases rapidly. This might be because attracting new contributors is difficult as they can be affected by both social and technical barriers [29], even though the ASFI community has established a set of specific rules that emphasize providing mentorship to newcomers [30]. Based on this finding, we believe that the more active an ASFI project is in the early stage (e.g., with more active voters), the greater the probability of it being graduated.

Figure 5 shows the distribution of the ratio of change intervals on *active_voters* to incubation months, where the *i-th* element represents the proportion of projects with a ratio of change intervals to incubation months. We observe that there are some differences in the distribution for graduated and retired projects. The majority of graduated projects have a change interval representing a small percentage of their total incubation time. Approximately 39.91% and 35.43% of projects have a ratio of change intervals to incubation time within 0% to 10% and 10% to 20%, respectively. This suggests that most graduated projects experience relatively short change intervals. In contrast, retired projects have a different distribution, with the highest proportion (43.86%) having a ratio of change intervals to incubation time within the 20% to 30% range, followed by 29.82% in the 10% to 20% range and 19.30% in the 0% to 10% range. Retired projects tend to have longer change intervals relative to their incubation time when compared to graduated projects.

Finding. There are indeed episodic changes in the voting process of projects during their incubation. Graduated projects tend to have shorter and fewer change intervals than retired projects, which may be indicative of a more consistently high activity level, which also means active project management, efficient collaboration, and timely issue resolution in graduated projects.

5 DISCUSSION

5.1 Case Study

To better understand the relationship between the Apache voting process and the final outcome of the ASFI projects, we further perform two case studies with different outcomes:

Case 1: *Ariatosca* is a project that offers an easily consumable Software Development Kit (SDK) and a Command Line Interface (CLI) to implement topology and orchestration specifications of

cloud applications. It remained in the incubator for about 23 months. During its incubation period, 11,657 emails including 2655 commits emails and 9002 developers emails are generated. Considering these data, *Ariatosca* is a very active project, yet it had a high probability to be graduated [34]. However, it finally retired.

By analyzing its voting activities, we find that *Ariatosca* starts the first voting event to publish its first release in the 11th month of the incubation. In the next month, the project members voted for the second release, and in the 17th month the third one. We noticed that the *Ariatosca* community rarely used the voting process for decision-making although they often communicated by the mailing list. After the voting event for the third release, the activities including commits and communications suddenly decreased a lot. Six months later, after a long time of inactivity, they decided to vote for the retirement of the project. The voting email said "Given that the project has not seen any activity over the past 5 months and all of the active committers have no time for the project - this is a call for Vote to retire the AriaTosca podling.".

It appears that *Ariatosca*'s retirement was directly caused by a lack of activity in the last month of the incubation. However, the root cause behind the community's inactivity might be that the decisions in *Ariatosca* are not made in a transparent and democratic manner. The Project Management Committee (PMC) of the project mainly made decisions on their own, and very few voting events are cast to ask the opinions of other project members. Based on this case, we believe that the voting process is important to the sustainability of ASFI projects.

Case 2: *Flex* is a highly productive, OSS application framework for building and maintaining expressive web applications that deploy consistently on all major browsers, desktops, and devices (including smartphones, tablets, and tv). It stayed in the incubator for 12 months. During the incubation period, 20,175 emails including 2,934 commits emails and 17,241 dev emails are generated, indicating that the project is very active.

In terms of the voting metrics, *Flex* is also active since there are approximately 80 voting events and 1,500 votes cast by its project members. The purposes of voting vary from determining the logo color to the branching strategy and project release. In the voting for graduation, 62 members were involved and the vote eventually passed. Based on the findings of our study, a project with active voting participation is more likely to be graduated

During the incubation period, the *Flex* project experienced a stable trend in voting activities. Only a single episodic change is identified for *total_votes*, *vote_times*, and *active_voters*, respectively. As the project experienced fluctuations in voting activities, the project members were able to quickly adapt to the changes and maintain a high level of engagement and participation.

In summary, this case underscores the importance of an active and transparent voting process in fostering a collaborative and engaged community, which ultimately contributes to the long-term success and sustainability of open-source software projects.

5.2 Implications

Based on the findings of our study, we discuss some implications for different roles in the OSS community.

For project maintainers: As an effective mechanism for community governance, voting can help project maintainers foster community-wide participation in decision-making. Project maintainers can leverage voting to accomplish various tasks (e.g., code modification and procedural issues) to enhance the community's engagement. Additionally, according to the findings of RQ3, voting at the beginning of the incubation time rather than after a period of operation is much more useful in improving the sustainability of a project. Hence, Project maintainers need to make the most of voting mechanisms as much as possible at the beginning of the incubation time. Based on the findings of RQ1 and RQ2, we find that the votes from mentors are important. Thus, project maintainers should try to attract more mentors to participate in voting, e.g., frequently asking them questions and listening to their opinions. Finally, if there are many inconsistent opinions among the voting results, project maintainers need to pay attention and conduct an in-depth analysis to identify the reasons for this result.

For project members: When considering joining an open-source community, project members should pay attention to the voting mechanisms in place. A well-functioning voting system is indicative of a project's potential for success and sustainability. By observing the community's voting process, members can assess the level of transparency and democratic decision-making which can help them make informed choices about which project to join. When given the opportunity to participate in voting, project members should seize the chance to voice their opinions and contribute valuable insights to the project. Engaging in the voting process not only allows members to influence project decisions but also demonstrates their commitment to the project's success. Moreover, active participation in voting can help members gain a deeper understanding of the project's goals and challenges, enabling them to contribute more effectively. Project members can benefit from observing and learning from the voting behavior of mentors within the community. By paying attention to how mentors express their opinions and make decisions through voting, members can gain insights into effective decision-making and communication strategies.

For researchers: Researchers can implement a tool or prediction model to monitor the voting process in an OSS community. Such a tool can evaluate the sustainability of the community by analyzing voting metrics and detecting episodic changes. By tracking and analyzing the voting activities, researchers can identify patterns and trends that may impact the project's success and help the community implement strategies to improve decision-making and overall project management. Studying the relationship between voting and other factors in open-source communities (e.g., social-technical factors or performance metrics) can enhance our understanding of their interactions. By studying these relationships, researchers can gain a deeper understanding of the complex dynamics within OSS projects and identify ways to enhance community collaboration and project performance. Furthermore, exploring how to optimize voting mechanisms can provide further insights into the open-source ecosystem. By examining the characteristics of successful voting systems and identifying barriers to participation, researchers can propose new strategies and techniques to foster inclusive decision-making processes that strengthen the overall health and sustainability of open-source projects.

5.3 Threats to Validity

Internal validity Our study mainly focuses on the voting process in the ASFI projects, which is just one aspect of the whole project. There may be other factors influencing the success or failure of projects in incubators that were not accounted for in the study, such as some social-technical metrics. Then, although we use regression models in our study, we do not use them to predict the project sustainability. Deep learning models may have an outstanding predictive ability for the outcome of the ASFI projects. Errors or omissions in the data collected from Apache Incubator mailing lists could also affect our analysis results.

External validity The dataset is based on the projects specifically in the ASF Incubator which have certain labels to measure sustainability. So, the findings may not generalize to all open-source projects or accurately represent trends in the broader open-source community.

6 RELATED WORK

The existing literature provides a number of theories and measures that have investigated OSS community governance and sustainability, such as [1, 4, 6, 8, 16, 20, 23, 25, 28, 32, 34, 35, 38], Among them, some of the studies focus on the impact of technical aspects on the sustainability of OSS projects, and some mainly analyze the social aspects, other than that many researchers combine the two aspects together to perform a more comprehensive study. Nyman et al. [23] analyzed the code forking mechanism and further explained the impact of governance on project sustainability. Mezouar et al. [12] focused on the features of social communications in Github Pull Request and explored the corresponding impact on the project's sustainability. They also used a regression model to identify some influential metrics. Capra et al. [6] argued that the governance of software projects is a process to help developers engage themselves better in the projects. O'Mahony and Ferraro [25] studied the technical features of the projects and made further analysis of the projects' governance policies. Yin et al. [34] use a social-technical network to analyze the combined chemical reactions of the two aspects and give a comprehensive analysis of the impact of social and technical features on the sustainability of the projects, providing the ASFI projects are labeled as sustainable, they also build a deep learning model to predict the label with the time series data. In another study, they proposed the episodic change theory and detected the episodic changes and corresponding change intervals (CI) in social-technical aspects [35], and further analyzed the relationship between CI and governance policies and regulations. A study compared the Apache Incubator to the Eclipse Incubator and proposed a set of best practices for applying this kick-off approach in newborn OSS projects [11]. The ability to attract newcomers is viewed as a sign of project sustainability. Therefore, some studies analyze the "good first issue" as it may make the project more friendly for newcomers [2, 3, 18]. Some studies focus on the Long-Term Contributors (LTC) of a project as LTCs can improve project sustainability, by predicting whether newcomers will become LTCs of a project [5, 14]. Additionally, some work analyzes developer turnover and its impact on project sustainability [15, 33]. Cortazar et al. [19] discussed the problem of developer turnover in software development projects and the resulting productivity losses with a methodology based on the amount of code contributed by a developer to measure the effect of turnover on software projects. Another study examined the impact of four factors on the behavior of developers to stay in a project since it may make the project more sustainable [21]. Valiev and Marat [31] explored the external factors in OSS sustainability, the mechanisms behind them, and proposed tools to make certain risk factors more visible. Though our work focuses on the different aspects from the previous work, we use some of the theories and methods to do the analysis as described in Section 4.

7 CONCLUSION

We present a novel perspective on the factors influencing open source project sustainability by analyzing data that include extrinsic sustainability labels from projects within the Apache Incubator. By focusing on the decision-making process, the study reveals that the voting activity differs significantly between graduated and retired projects, suggesting that transparent and democratic decision-making policies contribute to the long-term sustainability of projects. We also identify some influential metrics that can be useful for project members, especially mentors of the project to better contribute to the project. Practitioners can improve their decision-making abilities, organizational management and their personal skills by positively impact the overall success and sustainability of open source projects. This analysis of the Apache voting process highlights the importance of fostering a collaborative and democratic environment, enabling projects to thrive and contribute to the broader open source ecosystem.

ACKNOWLEDGMENTS

This project is supported by the Fundamental Research Funds for the Central Universities (No. 226-2022- 00064), the National Key Research and Development Program of China (No. 2021YFB2701102), and the National Science Foundation of China (No. U20A20173, No. 61902344, and No. 62141222)

REFERENCES

[1] Muhammad Shakil Ahmad and Noraini Bt Abu Talib. 2016. Analysis of community empowerment on projects sustainability: Moderating role of sense of community. *Social Indicators Research* 129 (2016), 1039–1056.
[2] David Alderliesten. 2020. The Effect of "Good First Issue" Indicators upon Newcomer Developers: Identifying Improvements for Newcomer Task Recommendation. (2020).
[3] Jan Willem David Alderliesten and Andy Zaidman. 2021. An Initial Exploration of the "Good First Issue" Label for Newcomer Developers. In *Proc. of CHASE*. IEEE, 117–118.
[4] Arzoo Atiq and Arvind Tripathi. 2016. Impact of financial benefits on open source software sustainability. (2016).
[5] Lingfeng Bao, Xin Xia, David Lo, and Gail C Murphy. 2019. A large scale study of long-time contributor prediction for github projects. *Trans. Softw. Eng.* 47, 6 (2019), 1277–1298.
[6] Eugenio Capra, Chiara Francalanci, and Francesco Merlo. 2008. An empirical study on the relationship between software design quality, development effort and governance in open source projects. *Trans. Softw. Eng.* 34, 6 (2008), 765–782.
[7] Norman Cliff. 1993. Dominance statistics: Ordinal analyses to answer ordinal questions. *Psychological bulletin* 114, 3 (1993), 494.
[8] Kevin Crowston, Kangning Wei, James Howison, and Andrea Wiggins. 2008. Free/Libre open-source software development: What we know and what we do not know. *ACM Computing Surveys (CSUR)* 44, 2 (2008), 1–35.
[9] Patrick Dawson. 2014. Reflections: On time, temporality and change in organizations. *Journal of Change Management* 14, 3 (2014), 285–308.
[10] Isabel Drost-Fromm and Rob Tompkins. 2021. Open Source Community Governance the Apache Way. *Computer* 54, 4 (2021), 70–75.

[11] Juan C Duenas, Félix Cuadrado, Manuel Santillán, José L Ruiz, et al. 2007. Apache and Eclipse: Comparing open source project incubators. *IEEE software* 24, 6 (2007), 90–98.

[12] Mariam El Mezouar, Feng Zhang, and Ying Zou. 2019. An empirical study on the teams structures in social coding using GitHub projects. *Empirical Software Engineering* 24 (2019), 3790–3823.

[13] PH Ellaway. 1978. Cumulative sum technique and its application to the analysis of peristimulus time histograms. *Electroencephalography and clinical neurophysiology* 45, 2 (1978), 302–304.

[14] Vijaya Kumar Eluri, Thomas A Mazzuchi, and Shahram Sarkani. 2021. Predicting long-time contributors for GitHub projects using machine learning. *Information and Software Technology* 138 (2021), 106616.

[15] Matthieu Foucault, Marc Palyart, Xavier Blanc, Gail C Murphy, and Jean-Rémy Falleri. 2015. Impact of developer turnover on quality in open-source software. In *Proc. of ESEC/FSE*. 829–841.

[16] Jonas Gamalielsson and Björn Lundell. 2014. Sustainability of Open Source software communities beyond a fork: How and why has the LibreOffice project evolved? *Journal of Systems and Software* 89 (2014), 128–145.

[17] Amir Hossein Ghapanchi. 2015. Predicting software future sustainability: A longitudinal perspective. *Information Systems* 49 (2015), 40–51.

[18] Yuekai Huang, Junjie Wang, Song Wang, Zhe Liu, Dandan Wang, and Qing Wang. 2021. Characterizing and Predicting Good First Issues. In *Proc. of ESEM*. 1–12.

[19] D. Izquierdo-Cortazar, G. Robles, F. Ortega, and J.M. Gonzalez-Barahona. 2009. Using Software Archaeology to Measure Knowledge Loss in Software Projects Due to Developer Turnover. In *2009 42nd Hawaii International Conference on System Sciences*. 1–10.

[20] Zhixing Li, Yue Yu, Tao Wang, Gang Yin, Shanshan Li, and Huaimin Wang. 2021. Are you still working on this? An empirical study on pull request abandonment. *Trans. Softw. Eng.* 48, 6 (2021), 2173–2188.

[21] Bin Lin, Gregorio Robles, and Alexander Serebrenik. 2017. Developer turnover in global, industrial open source projects: Insights from applying survival analysis. In *Proc. of ICGSE*. IEEE, 66–75.

[22] Henry B Mann and Donald R Whitney. 1947. On a test of whether one of two random variables is stochastically larger than the other. *The annals of mathematical statistics* (1947), 50–60.

[23] Linus Nyman and Juho Lindman. 2013. Code forking, governance, and sustainability in open source software. *Technology Innovation Management Review* 3, 1 (2013).

[24] Mancur Olson. 2012. The logic of collective action [1965]. *Contemporary Sociological Theory* 124 (2012).

[25] Siobhán O'mahony and Fabrizio Ferraro. 2007. The emergence of governance in an open source community. *Academy of Management Journal* 50, 5 (2007), 1079–1106.

[26] Huilian Qiu. 2022. *Understanding and Designing Mechanisms for Attracting and Retaining Open-Source Software Contributors*. Ph. D. Dissertation. Carnegie Mellon University.

[27] Anirudh Ramchandran, Likang Yin, and Vladimir Filkov. 2022. Exploring Apache incubator project trajectories with APEX. In *Proc. of MSR*. 333–337.

[28] Bhuricha Deen Sethanandha, Bart Massey, and William Jones. 2010. Managing open source contributions for software project sustainability. In *2010 Technology Management For Global Economic Growth*. IEEE, 1–9.

[29] Igor Steinmacher, Tayana Conte, Marco Aurélio Gerosa, and David Redmiles. 2015. Social barriers faced by newcomers placing their first contribution in open source software projects. In *Proc. of CSCWS*. 1379–1392.

[30] Igor Steinmacher, Marco Aurelio Graciotto Silva, Marco Aurelio Gerosa, and David F Redmiles. 2015. A systematic literature review on the barriers faced by newcomers to open source software projects. *Information and Software Technology* 59 (2015), 67–85.

[31] Marat Valiev. 2021. *External Factors in Sustainability of Open Source Software*. Ph. D. Dissertation. Carnegie Mellon University.

[32] Liang Wang, Ying Li, Jierui Zhang, and Xianping Tao. 2022. Quantitative analysis of community evolution in developer social networks around open source software projects. *arXiv preprint arXiv:2205.09935* (2022).

[33] Steven G Westlund and John C Hannon. 2008. Retaining talent: Assessing job satisfaction facets most significantly related to software developer turnover intentions. *Journal of Information Technology Management* 19, 4 (2008), 1–15.

[34] Likang Yin, Zhuangzhi Chen, Qi Xuan, and Vladimir Filkov. 2021. Sustainability forecasting for apache incubator projects. In *Proc. of ESEC/FSE*. 1056–1067.

[35] Likang Yin, Xiyu Zhang, and Vladimir Filkov. 2023. On the Self-Governance and Episodic Changes in Apache Incubator Projects: An Empirical Study. (2023).

[36] Marcelo Serrano Zanetti. 2012. The co-evolution of socio-technical structures in sustainable software development: Lessons from the open source software communities. In *Proc. of ICSE*. IEEE, 1587–1590.

[37] Marcelo Serrano Zanetti, Ingo Scholtes, Claudio Juan Tessone, and Frank Schweitzer. 2013. Categorizing bugs with social networks: a case study on four open source software communities. In *Proc. of ICSE*. IEEE, 1032–1041.

[38] Zhiwen Zheng, Liang Wang, Jierui Zhang, Baihui Sang, and Xianping Tao. 2022. Measure the Diversity of Open Source Software Projects' Forks with Fork Entropy. *arXiv preprint arXiv:2205.09931* (2022).

[39] Minghui Zhou and Audris Mockus. 2014. Who will stay in the floss community? modeling participant's initial behavior. *Trans. Softw. Eng.* 41, 1 (2014), 82–99.