Comparing Emperical Methods

ECE1785: Assignment 1 - Comparison of Methods

Amos Hebb, Yilong Wang, Zhenyue Yu
University of Toronto

I. Introduction

Two papers, one a multiple case study, one a computational analysis followed by a preliminary analysis of a dataset, both study open source developers. These papers are used as examples of both quantitative and qualitative research. We then compare and contrast these approaches.

II. How to Break an API

The first paper we will consult is “How to Break an API” [1], a 2016 multiple case study of three ecosystems philosophies toward change.

A. Method

A multiple case study, interviewing 28 developers in each of three ecosystems. Developers were recruited with recent, relevant experiences in similar roles contributing to packages with multiple upstream and downstream dependencies. The three ecosystems studied are Eclipse, R/CRAN, and Node.js/npm. Semi-structured interviews focused on personal practices and experiences negotiating upstream and downstream dependencies. Interviews were recorded, transcribed, then coded. Findings were validated with Dagenais and Robillard’s [2] methodology to check fit and applicability as defined by Corbin and Strauss [3].

B. Research Questions

- How do developers make decisions about whether and when to perform breaking changes?
- How do they mitigate or delay costs for other developers?
- How do developers react and manage change in their dependencies?
- How do policies, tooling, and community values influence decision-making?

C. Evidence used to Reach Conclusion

Patterns in coded interviews, combined with policies, public discussions, meeting minutes, and tools in each ecosystem were compared to find patterns. This revealed that while communities have similar reasons for breaking changes, and that developers are aware of the costs changes cause to users, community values influence how costs are shifted which explains differences observed among the ecosystems.

III. Need for tweet

The second paper we studied is “Need for tweet: How open source developers talk about their GitHub work on twitter“ [4] .

A. Method

In the second paper, the researchers built a database of the linkage between Twitter and GitHub accounts and performed an ethnography case study on how different developers with different roles associated with specific projects tend to have different tweeting patterns. They proposed a heuristic computational method to link software developers’ GitHub accounts to their Twitter accounts. Then they performed a qualitative analysis of the content of a sample of GitHub users’ tweets.

B. Research Question

- How to link developers cross-platform?
- How are the tweet contents correlate with the role of the open source developers on GitHub projects?

C. Evidence used to reach conclusion

In the validation process, the authors filtered out obvious false positives and conducted a random sampling check to show that their database has high validity, over 85Based on the final dataset, the author categorized six major emerging themes with respect to five main GitHub project roles. This paper used the quantitative analysis on a random sample to demonstrate the distribution of emerging themes across the different role strata.

IV. When each method is appropriate

A. Qualitative

Qualitative research is best for trying to understand broader concepts or behaviour. 'Break an API', in the words of the participants, *"brings a structure and coherence to issues that I was loosely aware of, but that are too rarely the centre of focus"* [1]. The use of a qualitative approach can also be beneficial when researchers are trying to classify or categorize something specific.

B. Quantitative

Quantitative research is best for trying to answer a specific question, especially if the property being studied is easily measured. In the case of 'Need for Tweet', the authors used quantitative analysis to the distribution of Twitter content across the different developer roles. Besides that, public contributions to open-source projects and public Twitter profiles can be counted, allowing for community-level analysis.

Another benefit of quantitative research is that artifacts are produced that are not only useful for reproducibility but other perhaps unrelated research.

V. Weaknesses of each method

A. Qualitative

High-touch studies limit the sample sizes, and those who are open to being interviewed may have other selection biases. There may also be social desirability biases, in 'Break an API' all respondents indicated that they care deeply about downstream users.

B. Quantitative

Measuring social phenomena that are inherently subjective is challenging. In 'Need for Tweet', tweets were binned by theme, and users were binned by role. Reality is far more nuanced, the line between 'info share' and 'ad' is elusive at best, and developers may serve multiple or change roles on projects. This can result in a narrow focus on phenomena, and subjects, that are easily measured.

Combined with modern data mining techniques, even a well-designed sampling method may introduce subtle systematic errors. Large sample sizes also cause statistically significant results to emerge where patterns may not exist. Even in the statistically simple "Need for Tweet", sample sizes over 100 thousand were paired with old rules of thumb. A 95% confidence interval is used to select sample sizes without accounting for a loss of sensitivity due to only 40 of 50 randomly sampled links being correct.

VI. Conclusion

Both qualitative and quantitative studies are important parts of the empirical method. 'Break an API' exhibits the rich understanding that comes from interviewing individuals with similar roles in different communities. It may be difficult to replicate due to the nuanced answers and small sample sizes. 'Need for Tweet' is easily reproduced as these data were made public. It provides insights into population-level statistics that could not be analyzed without computationally combining these previously unlinked data. It focuses narrowly and bins tweets and developers into artificially discrete buckets necessary for statistical analysis. With a better understanding of both approaches, we are now prepared to appropriately apply the correct study methodology to our questions and analyze and structure the results appropriately.

References

[1] C. Bogart, C. Kästner, J. Herbsleb, and F. Thung, "How to break an api: cost negotiation and community values in three software ecosystems," in *Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering*, 2016, pp. 109–120.

[2] B. Dagenais and M. P. Robillard, "Creating and evolving developer documentation: understanding the decisions of open source contributors," in *Proceedings of the eighteenth ACM SIGSOFT international symposium on Foundations of software engineering*, 2010, pp. 127–136.

[3] J. Corbin and A. Strauss, *Basics of qualitative research: Techniques and procedures for developing grounded theory*. Sage publications, 2014.

[4] 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.