TU/e EINDHOVEN UNIVERSITY OF TECHNOLOGY

Crowdsourced knowledge catalyzes software development (Extended abstract)

Citation for published version (APA):
Vasilescu, B. N., Filkov, V., & Serebrenik, A. (2013). Crowdsourced knowledge catalyzes software development (Extended abstract). In T. Mens, M. Claes, M. Goeminne, & S. Drobisz (Eds.), *12th Belgian-Netherlands Software Evolution Seminar (BENEVOL'13, Mons, Belgium, December 16-17, 2013)* (pp. 58-60). Université de Mons.

Document status and date:
Published: 01/01/2013

Document Version:
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

Download date: 16. Nov. 2023

Crowdsourced Knowledge Catalyzes Software Development

Bogdan Vasilescu
Eindhoven University of Technology, The Netherlands
b.n.vasilescu@tue.nl

Vladimir Filkov
University of California, Davis, USA
filkov@cs.ucdavis.edu

Alexander Serebrenik
Eindhoven University of Technology, The Netherlands
a.serebrenik@tue.nl

I. Introduction

Developers create and maintain software by standing on the shoulders of others [1]: they reuse components and libraries, and go foraging on the Web for information that will help them in their tasks [2]. For help with their code, developers often turn to programming question and answer (Q&A) communities, most visible of which is StackOverflow. To engage its participants to contribute more, StackOverflow employs gamification [3]: questions and answers are voted upon by members of the community; the number of votes is reflected in a person's *reputation* and *badges*; in turn, these can be seen as a measure of one's expertise by peers and potential recruiters [4] and are known to motivate users to contribute more [3], [5].

The analogy of StackOverflow as an effective educational institution asserts itself then. The extended effect of education, beyond the immediate edification, is to accelerate or catalyse societal advances. Does StackOverflow have the same effect on software development communities? The connection between developer productivity and their using of StackOverflow is not well-understood. On the one hand, StackOverflow is known to provide good technical solutions [6] and to provide them fast [7], to the extent that closer integration between Q&A websites and modern IDEs is now advocated [8], [9]. On the other hand, as an exponent of social media, using StackOverflow may lead to interruptions impairing the developers' performance [1], especially when gamification is factored in.

In a recent paper [10] we investigated the interplay between asking and answering questions on StackOverflow and committing changes to open-source GitHub repositories. This extended abstract summarises our main findings. GitHub is arguably the largest social coding site, hosting more than three million software projects maintained by over one million registered developers. The two platforms overlap in a knowledge-sharing ecosystem (Figure 1): GitHub developers can ask for help on StackOverflow to solve their own technical challenges; similarly, they can engage in StackOverflow to satisfy a demand for knowledge of others, perhaps less experienced than themselves, or to compete in the "game" to achieve higher reputation. By identifying GitHub users active on StackOverflow and studying their activities on both platforms, we can study if a connection exists between their participation in StackOverflow and their productivity on GitHub. GitHub users are a mix of novice and professional programmers [11]. While it is known that foraging is common for novices and experts alike [2], their *diets* are different [12], with potentially different impact on their performance. Is participation in StackOverflow related to productivity of GitHub developers? Is it more beneficial for some groups of developers than for others? Do the StackOverflow activities impede GitHub commit activities or do they accelerate them?

II. Experimental Setup

We integrated data from two sources: StackOverflow (as part of the Stack Exchange data dump released in August 2012, containing information about 1,295,622 registered users) and GitHub (from GHTorrent [13], a service that gathers event streams and data from GitHub, containing information about 397,348 users and 10,323,714 commits from the July 2011 to April 2012 period).

A key step in our process was merging the GitHub and StackOverflow datasets, i.e., identifying those contributors which were active on both platforms. Merging aliases used by the same person in different software repositories is a well-known problem [14]–[17]. We followed a conservative approach to identity merging and made use of email addresses, present in the GitHub dataset but obscured using an MD5 hash in the StackOverflow one. We decided to merge (i.e., link) a GitHub and a StackOverflow user if the computed MD5 hash of the former's email address was identical to the MD5 email hash of the

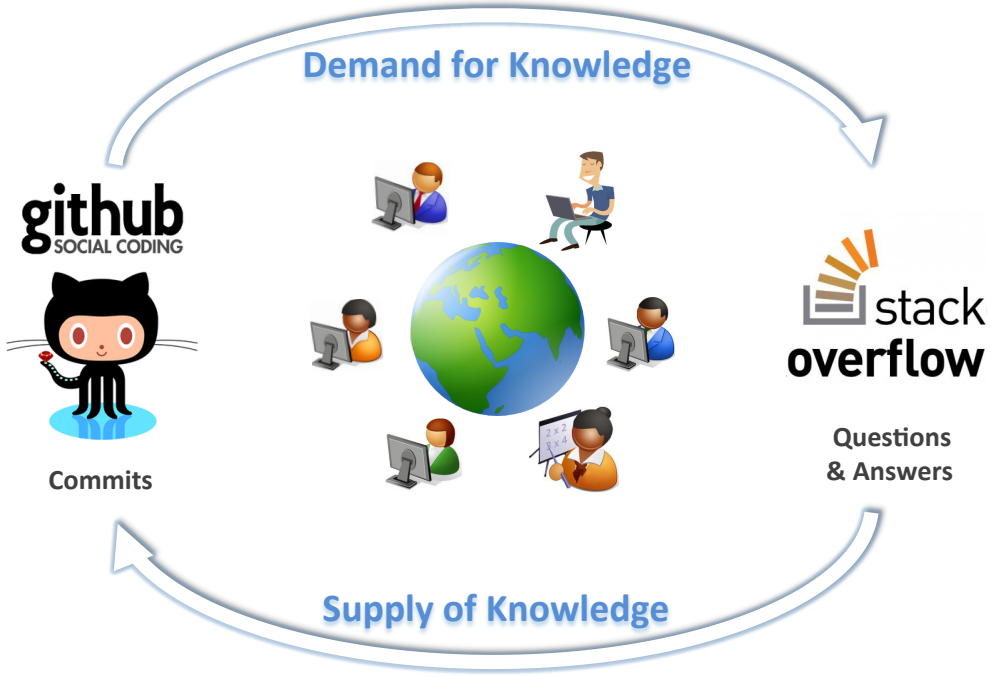

Fig. 1: Demand and supply of knowledge between source code and Q&A.

latter, resulting in approximately one quarter of the GitHub users (23.6%, or 93,771) being linked to StackOverflow. Only 46,967 of these (or 11.8% of the GitHub dataset) asked or answered at least one question on StackOverflow between July 2011 and April 2012.

III. Findings

A. *StackOverflow Experts are Active GitHub Committers*

First, we focussed on differences in StackOverflow involvement of the GitHub developers. We found a *direct relationship between GitHub commit activity and StackOverflow question answering activity*: the more active a committer, the more answers she gives. In other words, highly productive committers tend to take the role of a "teacher" more actively involved in providing answers rather than asking questions. Similarly, the more active an answerer, the more commits she authors. In other words, top users on StackOverflow are "superstars" rather than "slackers": they don't just compete for reputation and badges, but are actually active (open-source) software developers.

In contrast, we found an *inverse relationship between GitHub commit activity and StackOverflow question asking activity*: active GitHub committers ask fewer questions than others; less active question askers produce more commits. Overall, these findings suggest that an activity-based ranking of StackOverflow contributors reflects one extracted from their open-source contributions to GitHub, increasing the confidence in the reliability of social signals based on StackOverflow (e.g., answering questions on StackOverflow can be seen as a proxy for one's commit activity on GitHub).

B. *Experts and Novices Have Different Working Rhythms*

Next, we studied whether the working rhythm of the GitHub contributors is related to their StackOverflow activities. We observed that individuals that tend to ask many questions distribute their effort in a less egalitarian way than developers that do not ask questions. No differences were observed between the work distributions for individuals grouped based on the number of answers given. In other words, developers who ask many questions on StackOverflow commit changes to GitHub in bursts of intense activity followed by longer periods of inactivity, i.e., they focus their attention at any given time. Specialization (or focus) of developers has also been noted previously in the context of activity types (e.g., coding versus translating) or files touched as part of a shared project [17], [18]. Therefore, *asking* questions on StackOverflow influences how developers distribute their time over commits on GitHub, while *answering* questions does not seem to have the same effect. We conjecture that this observation is due to developers learning from StackOverflow and committing their experiences to GitHub.

C. Crowdsourced Knowledge Catalyzes Software Development

Finally, we associated GitHub commits and StackOverflow questions and answers over time, in an attempt to understand whether activities in the two platforms show signs of coordination. We found that the rate of asking or answering questions on StackOverflow is related to the rate of commit activities in GitHub. In other words, despite interruptions incurred, for active GitHub developers StackOverflow activities are positively associated with the social coding in GitHub. Similar observations hold for active askers as well as individuals who have been involved in GitHub for sufficiently long time. Finally, StackOverflow activities accelerate GitHub committing also for the most active answerers as well as for developers that do not answer any questions at all.

REFERENCES

[1] M.-A. D. Storey, C. Treude, A. van Deursen, and L.-T. Cheng, "The impact of social media on software engineering practices and tools," in *FoSER*. ACM, 2010, pp. 359–364.

[2] J. Brandt, P. J. Guo, J. Lewenstein, M. Dontcheva, and S. R. Klemmer, "Two studies of opportunistic programming: interleaving web foraging, learning, and writing code," in *CHI*. ACM, 2009, pp. 1589–1598.

[3] S. Deterding, "Gamification: designing for motivation," *Interactions*, vol. 19, no. 4, pp. 14–17, 2012.

[4] A. Capiluppi, A. Serebrenik, and L. Singer, "Assessing technical candidates on the social web," *IEEE Software*, vol. 30, no. 1, pp. 45–51, 2013.

[5] A. Anderson, D. P. Huttenlocher, J. M. Kleinberg, and J. Leskovec, "Discovering value from community activity on focused question answering sites: a case study of Stack Overflow," in *KDD*. ACM, 2012, pp. 850–858.

[6] C. Parnin, C. Treude, L. Grammel, and M.-A. Storey, "Crowd documentation: Exploring the coverage and the dynamics of API discussions on Stack Overflow," Georgia Institute of Technology, Tech. Rep., 2012.

[7] L. Mamykina, B. Manoim, M. Mittal, G. Hripcsak, and B. Hartmann, "Design lessons from the fastest Q&A site in the west," in *CHI*. ACM, 2011, pp. 2857–2866.

[8] A. Bacchelli, L. Ponzanelli, and M. Lanza, "Harnessing Stack Overflow for the IDE," in *RSSE*. IEEE, 2012, pp. 26–30.

[9] J. Cordeiro, B. Antunes, and P. Gomes, "Context-based search to overcome learning barriers in software development," in *RAISE*, 2012, pp. 47–51.

[10] B. Vasilescu, V. Filkov, and A. Serebrenik, "StackOverflow and GitHub: Associations between software development and crowdsourced knowledge," in *Proceedings of the 2013 ASE/IEEE International Conference on Social Computing*. IEEE, 2013, pp. 188–195.

[11] L. A. Dabbish, H. C. Stuart, J. Tsay, and J. D. Herbsleb, "Social coding in GitHub: transparency and collaboration in an open software repository," in *CSCW*. ACM, 2012, pp. 1277–1286.

[12] B. Evans and S. Card, "Augmented information assimilation: social and algorithmic web aids for the information long tail," in *CHI*. ACM, 2008, pp. 989–998.

[13] G. Gousios and D. Spinellis, "GHTorrent: Github's data from a firehose," in *MSR*. IEEE, 2012, pp. 12–21.

[14] C. Bird, A. Gourley, P. T. Devanbu, M. Gertz, and A. Swaminathan, "Mining email social networks," in *MSR*. ACM, 2006, pp. 137–143.

[15] M. Goeminne and T. Mens, "A comparison of identity merge algorithms for software repositories," *Science of Computer Programming*, vol. 78, no. 8, pp. 971–986, 2011.

[16] E. Kouters, B. Vasilescu, A. Serebrenik, and M. G. J. van den Brand, "Who's who in Gnome: Using LSA to merge software repository identities," in *ICSM*. IEEE, 2012, pp. 592–595.

[17] B. Vasilescu, A. Serebrenik, M. Goeminne, and T. Mens, "On the variation and specialisation of workload—a case study of the Gnome ecosystem community," *Empirical Software Engineering*, pp. 1–54, 2013.

[18] D. Posnett, R. D'Souza, P. Devanbu, and V. Filkov, "Dual ecological measures of focus in software development," in *ICSE*. IEEE, 2013, pp. 452–461.