

How the R Community Creates and Curates Knowledge: A Comparative Study of Stack Overflow and Mailing Lists

Alexey Zagalsky, Daniel M. German,
Margaret-Anne Storey, Carlos Gómez Teshima,
Germán Poo-Caamaño

2016

Abstract One of the effects of social media’s prevalence in software development is the many flourishing communities of practice where users share a common interest. These large communities use many different communication channels, but little is known about how they create, share, and curate knowledge using such channels.

In this paper, we report a mixed methods study of how one community of practice—the R software development community—creates and curates knowledge associated with questions and answers (Q&A) in two of its main communication channels: the R tag in Stack Overflow and the R-users mailing list. The results reveal that knowledge is created and curated in two main forms: participatory, where multiple users explicitly collaborate to build knowledge, and crowdsourced, where individuals primarily work independently of each other. Moreover, our study reveals participation patterns, showing the existence of prolific contributors: users who are active across both channels and are responsible for a large proportion of the answers, serving as a bridge of knowledge.

The key contributions of this paper are: a characterization of knowledge artifacts that are exchanged by this community of practice; the reasons why users choose one channel over the other; and insights on the community participation patterns, which indicate an evolution of the community and a shift from knowledge creation to knowledge curation. Finally, this paper enumerates a set of recommendations to assist practitioners in the use of multiple channels for Q&A.

1 Introduction

The emergence and adoption of socially enabled tools and channels (e.g., GitHub, Stack Overflow, mailing lists) has fostered the formation of large *communities of practice* where users share a common interest, such as programming languages, frameworks, and tools [20]. These communities rely on many different communication

channels, but little is known about how they create, share, and curate knowledge using such channels.

One prominent community of practice is the group that has formed in support of the R programming language, an open source project without commercial backing that relies heavily on its rapidly growing and highly heterogeneous software development community. The R community plays an important role in diffusing the R language: users have access to numerous resources for learning the language and receiving help, such as mailing lists, blogs, books, online and offline courses, and question & answer sites (e.g., Stack Overflow). While the R community benefits from this vast and rich corpus of knowledge, it also drives the creation and curation of the information.

Without a single entity directing and controlling it, the R language has grown organically from its community. Similar to other communities of practice, knowledge is exchanged and curated in many communication channels, and two particular communication channels are at the center of this process: the *R-help mailing list* and *Stack Overflow*. The R-help mailing list was created to assist those using the language, and while Stack Overflow is not specifically oriented towards R, its section dedicated to R (the R tag) has grown rapidly¹.

Stack Overflow has revolutionized the way programmers seek knowledge [12, 23], assuming the role of a capable “expert on call” that is able—and willing—to answer questions of any level of difficulty about any programming technology (R included). Stack Overflow’s gamification features guarantee that enthusiastic experts will answer questions, often within minutes of being posted [13]. Equally important is the ability of Stack Overflow’s users to curate the knowledge being created, making sure that the best answers surface to the top and become a valuable asset to those seeking an answer now or in the future. Stack Overflow has become a popular and effective tool for creating, curating, and exchanging knowledge, including knowledge about the R language.

One would expect that the traffic on the R-help mailing list would begin to fizzle as Stack Overflow popularity increased. If Stack Overflow is so effective at matching those who seek knowledge with those that have it, doesn’t that obviate most of the need for the R-help mailing list? Yet that does not appear to be the case as the R-help mailing list has maintained a steady level of activity, implying that it is still an important resource for the R community. It even appears as if the mailing list and Stack Overflow complement each other.

There are obvious inherent differences between both communication channels. On the one hand, mailing lists unite users by subscription, creating a tight community, but their content lacks organization, except for the natural structure provided by email metadata (e.g., subjects, threading, authors, dates), and they are not optimized for long-term storage and retrieval. On the other hand, Stack Overflow’s community is not as tight as the R-help mailing list community, but the channel is optimized for the curation and long-term storage of knowledge. However, little is known about the differences in how people use both communication channels, such as how the types of questions and answers sought in one channel compare to the other, why users choose

¹ <http://www.r-bloggers.com/r-is-the-fastest-growing-language-on-stackoverflow/>

one channel over the other, why some users participate in both channels, and how participants perceive each communication channel.

In this paper, we empirically compare how knowledge, specifically knowledge manifested as questions and answers, is sought, shared, and curated on both the R-help mailing list and Stack Overflow. We also review the participation patterns of users in both communities. Our research employed a mixed methods *exploratory case study* methodology to answer the following research questions:

- RQ1.** What types of knowledge artifacts are shared on Stack Overflow and the R-help mailing list within the R community?
- RQ2.** How is the knowledge constructed on Stack Overflow and the R-help mailing list?
- RQ3.** Why do users post to a particular channel and why do some post to both channels?
- RQ4.** How do users participate on both channels over time?
- RQ5.** Are there significant differences in participation activity between community users?

By mining archival data, we identified and categorized the main types of knowledge artifacts contained on the R-help mailing list and in Stack Overflow messages (RQ1). The emerging categories form a *typology* (see Table 3) that allows researchers to study and characterize Q&A knowledge dissemination within a community of practice. We used this typology to study how knowledge is constructed and shared on Stack Overflow and the R-help mailing list. We found that these channels support two distinct approaches for constructing knowledge—*participatory knowledge construction* and *crowd knowledge construction*—however, each channel supports them differently (RQ2). Our findings indicate that participatory knowledge construction is more prevalent on the R-help mailing list, while crowd knowledge construction is more prevalent on Stack Overflow.

We found that some contributors are active on both channels. As a result, we conducted a survey to investigate the benefits they gain by doing so (RQ3). But beyond that, we wanted to examine how participation differs between Stack Overflow and the R-help mailing list over time and how long users participate on the two channels (RQ4). Additionally, we wanted to understand the behavior and *participation patterns* of the contributing users (RQ5). We focused on several sets of contributors: those who rarely contribute, the top contributors, and those who contribute to both channels. Our results show that a great majority of participants are fleeting and a small number of individuals are responsible for most answers. Furthermore, our findings indicate that both channels are reaching maturity: for the R-help mailing list, this means a steady flow of questions; for Stack Overflow, there is a continuous decrease of questions with a positive score (number of positive votes minus negative votes), hinting to the fact that, as time progresses, the most sought after questions have been already asked.

We conclude the paper by providing recommendations for using the different communication channels, and discussing how channel affordances and community

rules (e.g., topic restriction and gamification) influence knowledge construction and curation.

2 Background

We begin with an overview of the R community and detail the two main channels its users use to ask and answer questions.

2.1 The R Community of Practice

The R project² was born in 1993 as a free and open source programming language and software environment for statistical computing, bioinformatics, and graphics [9]. The R community is composed of: (1) *R-core*, a team of 20 software developers that maintain and evolve the R language; and (2) *Periphery*, which includes everyone else (language users and package developers).

The R community is an eclectic open source community that goes beyond software development and includes biologists and statisticians with no or limited programming experience. Its entire history of mailing list communication is archived and publicly available. The R community has also been the subject of extensive research in community evolution [7, 22] and the interplay between channels [23].

R's popularity has continuously increased over the years. IEEE Spectrum recently ranked R as the 5th most popular language³, implying that the R community is healthy and continues to grow at a fast rate.

Our study focused on the analysis of Stack Overflow and the R-help mailing list, two channels in the R community. We chose them because they are the main channels that provide Q&A support to the community.

2.1.1 The R-help mailing list

There are several mailing lists to help R community users solve programming problems with the R language: *R-help*, *R-package-devel*, *R-devel*, *R-packages*, *R-announce*, and *Bioconductor*. However, the R-help mailing list is the main channel for discussing problems and solutions using R. Other messages are also encouraged, such as documentation, benchmarks, examples, and announcements.

The R-help mailing list used to be the main communication channel for asking and answering questions within the R community, but a significant number of users migrated to Stack Overflow [23]. Despite the reduced number of users, the R-help mailing list is still very active—on average, a subscriber may receive approximately 25 emails a day (as of Oct 2016).

² <https://www.r-project.org/>

³ <http://spectrum.ieee.org/computing/software/the-2016-top-programming-languages>

2.1.2 Stack Overflow

In contrast to the R-help mailing list, Stack Overflow incorporates a rich visual and user-friendly interface with social media and gamification features. The social aspect of the Website improves participation and provides strong support for creating and sharing knowledge as well as encouraging informal mentorship [10,20]. Meanwhile, Stack Overflow's gamification features provide reputation points and badges to reward user participation and earn them points that enable functionality inside the site. It has been reported that Stack Overflow's gamification mechanisms boost participation [22] and enable mutual assessment [16].

2.2 Stack Overflow vs. Mailing Lists

Software development is a knowledge-building process [14]. Due to the emergence of socially enabled tools and channels and the formation of communities of practice [20], it is important to understand how knowledge is created and shared within these communities. In our study, we focus on knowledge in the form of questions and answers within the R community.

As part of a study on the transition to gamified environments, Vasilescu [22] examined the popularity of Stack Exchange (including the Stack Overflow R tag) and mailing lists within the R community. He found that the number of message threads has decreased on the R-help mailing list since 2010, while the number of R-related questions asked on the Stack Exchange network has increased. Our study also examined Stack Overflow's R tag and the R-help mailing list, but we aimed to understand the *knowledge types* used. This allowed us to characterize the different knowledge seeking and sharing approaches on each channel. Vasilescu also examined the *difference in activity* between contributions made by users active on both channels and users focused on a single medium. While we also found users of the R community that were active on both channels, we first aimed to understand *why users post to a particular channel*, and then understand *user participation activity over time*.

Similar to Vasilescu, Squire [17] studied a project's *transition to the Stack Overflow gamified channel*. She focused on examining whether four software projects that moved from mailing lists to Stack Overflow showed improvements in terms of developer participation and response time. She found that all four projects showed improvements on Stack Overflow compared to mailing lists, however, she also found that several projects have moved back to using mailing lists despite achieving these improvements. In our study, we found that both channels have knowledge support for question and answers, however, there are important differences between the two channels. For example, Stack Overflow's competitive environment creates an incentive to be the first to answer rather than improve other answers and participate in discussions.

3 Methodology

As mentioned, the main goal of this work is to empirically compare how knowledge, specifically knowledge in question-and-answer (Q&A) form, is sought, shared, and curated on both the R-help mailing list and the R tag on Stack Overflow. We used a mixed methods *exploratory case study* methodology [6, 15] to answer our research questions (presented in Section 1).

This study employed two research methods over three phases: we *mined archival data* and conducted a *qualitative survey*. In the first phase, we randomly sampled and iteratively coded⁴ questions in both channels to characterize the types of discussions that occur. This process was continued until we reached saturation, which amounted to 400 threads in each channel. In the second phase, we surveyed users of the R community to validate our interpretation of the results from the previous phase. And in the third phase, we extended the collected data set up to September 2016 and focused on investigating participation patterns on both channels.

3.1 Phase I: Mining Archival Data

We mined data from the public archives of both the R-help mailing list and Stack Overflow. The R-help mailing list archive started in 1997, while the archives for Stack Overflow started in 2008 (when it was created). To make the data sets comparable, we analyzed both datasets from September 2008 until December 2013, a period of time that both channels were available. For Stack Overflow, we obtained a data dump file from their Website—Stack Exchange releases a new data dump from all their Websites every three months⁵. For the R-help mailing list data, we retrieved MBOX files of the archives from the R-help Website.

To answer RQ3, we also needed the ability to compare email addresses between both channels. However, Stack Exchange stopped providing information about email addresses—the last Stack Exchange dump file to contain email addresses as MD5 hashes was released in September 2013, so this meant we did not have complete data for October, November, and December 2013. To remedy this gap, we used the data dump file from September 2014, but updated the `users` table with the hashes taken from the September 2013 dump file for those IDs that were identical in both data sets. If a user in the 2013 data file did not exist in the 2014 data, we did not examine them. From Stack Overflow, we retrieved all R-related data by selecting only messages that contained the R tag (`r`) or its two synonyms⁶, (`rstats` and `r-language`).

To determine which users were active in both channels, we compared the MD5 hashes of the email addresses extracted from the R-help mailing list and Stack Overflow. If there was a match, we associated the R-help and Stack Overflow user accounts to the same person. Using this approach, we identified 1,421 persons who used both media channels.

⁴ Our sample data is openly available at <https://github.com/thechiselgroup/R-ML-and-StackOverflow>

⁵ <http://stackexchange.com/sites>

⁶ <http://stackoverflow.com/tags/r/synonyms>

To prepare the data, we used two different software tools: (1) To process the Stack Overflow data, we used a modified version of Sam Saffron’s application, So-Slow⁷. (2) To process the R-help mailing list data, we wrote our own mail mining application⁸. To ensure accurate results when processing the R-help mailing list, we followed a series of recommendations proposed by Bettenburg *et al.* [3]: extracted messages, removed duplicates, removed signatures, and reconstructed discussion threads. Table 1 depicts a summary of the data used for this study. Unsurprisingly, the R-help mailing list has more questions, answers, and users as it contains approximately ten years of additional data. Note that only Stack Overflow’s data contains “comments” information.

Table 1: Raw data collected for each channel during Phase I - up to September 2014.

Type	R-help	Stack Overflow
Questions	101,931	67,393
Answers	213,366	99,620
Comments	-	286,124
Different individuals	31,729	26,324

3.1.1 Data analysis process

We followed an inductive approach [15] to analyze the data from Stack Overflow and the R-help mailing list. To reduce the risk of bias [15], the analysis was conducted by two computer scientists with a background in qualitative data analysis. To answer RQ1 and RQ2, we selected 400 random threads from each channel. To answer RQ3, we focused on questions with identical subjects that were posted to both channels by the same author—we found and analyzed 79 such threads.

ID	Message	Channel	Question	Answer	Update	Comment	Flag	Resource	Knowledge construction	Memos	URL
1	MessageId: 1716012 Subject: Stopwatch function in R Date: 2009-11-11	SO	Environ ment	-	Expansion : Non- labelled	-	-	Official documentation;E xpand	Participatory	Software development questions Solving an error Should be flagged as Debugging Many answer	http://stackoverflow.com/questions/1716012
2	MessageId: 1340054801327-4633754.post Subject: [R] (1-le tters)	RH	Discrepa ncy	-	-	-	-	-	Crowd	Well explained question	http://r.89695.4.nabble.e.com/71340054801327-4633754-post1.html

Fig. 1: Example of data coding. Each row is a threaded message. Questions, comments, and answers are identified with the number on the first column. Columns in yellow (columns 4-10) contain the code for each message type. The last two columns contain the memos and URLs.

We used memoing, affinity diagrams, and a code book to support the data analysis process. We wrote reflective memos in a spreadsheet next to the applicable codes (see example in Fig. 1). These memos were used to create the codes and hypotheses about the relationships between concepts. We coded in multiple sessions, which allowed us

⁷ <https://github.com/SamSaffron/So-Slow>

⁸ Our tool is available at <https://github.com/cagomez/GTMail>

to refine the definitions in the code book iteratively. Each entry is associated with a title, a formal definition, an example, and notes from the researchers. For inter-rater reliability, we used the Cohen Kappa inter-rater agreement coefficient [19]. Although it is suggested that one should aim for coefficient values above 0.6 to obtain substantial results [11], based on our previous experience with this method [8], we aimed for 0.8 or above. We used this coefficient after each coding session as a way to trigger discussion and to further refine the codes if necessary. The emergent codes were fully saturated after reviewing 400 threads from each channel.

The analysis process required an *understanding of the context* surrounding each message. The process consisted of: (1) gathering the required information from each channel (i.e., the message analyzed, the relevant thread), and (2) mapping the messages from each channel to a specific knowledge type (see Section 4.1). The mapping was necessary as each channel contained a different data structure. We defined the following mappings between messages in both channels:

Question: The message is the first in the thread and contains the main question.

Answer: The message provides a solution to the main question of the thread.

Update: The message provides a modification to a question or answer made by the author of said question or answer.

Comment: The message offers clarification to a specific part of the question or answer.

Flag: The message requests attention from the moderator (e.g., repeated questions, spam, or rude behavior).

3.2 Phase II: Qualitative Survey

The analysis from Phase I revealed that some developers are active on both channels, and in some cases, even post the same questions. To further understand this phenomena and explore the perceived benefits of using one channel over the other, we conducted a survey with users of the R community⁹. To test and refine the questions, format, and tone, we piloted the survey twice. We promoted our survey on Twitter, Reddit, the R-help mailing list, and Meta Stack Exchange to reach users of both channels and minimize selection bias. However, our survey invitation on Stack Exchange was deemed off topic and deleted a few minutes later. In total, we received 37 responses, 26 of which were valid (invalid responses occurred if the session ended or the participant did not complete the survey).

3.3 Phase III: An Extended Quantitative Analysis

In this phase, we focused our analysis on the behavior and participation patterns of community users on the R-help mailing list and Stack Overflow. We extended our analysis to include archival data from September 2008 up to September 2016 for both channels.

⁹ A copy of the survey is available at <http://goo.gl/mxmH5J>

For Stack Overflow, we used Stack Exchange’s Data Explorer¹⁰ to query the data. The structured metadata allowed us to automatically classify each message as a question, an answer, or a comment.

For the R-help mailing list, we downloaded all the email threads and extracted the separate emails and corresponding metadata from each thread. As opposed to Stack Overflow, the R-help mailing list data doesn’t include metadata to indicate whether a message is a question or an answer. Thus, we used the following heuristic to classify the R-help messages: if a message contained an `In-Reply-To` or a `References` header, it was classified as an answer (239,604 responses); otherwise, a message was classified as a question (124,791 questions).

To study the participation behavior of different users, we assigned them a unique person identifier and performed a unification of identities. We extracted email addresses from the R-help mailing list data using the `From:` field, and using a conservative approach, identified messages sent by the same person but from different email addresses. As a result, the number of unique individuals identified on R-help was reduced from 36,600 to 31,729 (a reduction of 15%)—a similar number was reported by Vasilescu *et al.* [23]. Next, by using MD5 hashes of the email addresses extracted from the R-help mailing list and Stack Overflow, we identified 1,480 persons who used both media channels. We summarize the data collected during this phase in Table 2.

Table 2: Raw data collected for each channel during Phase III - up to September 2016.

Type	R-help	Stack Overflow
Questions	124,791	67,393
Answers	239,604	99,620
Comments	-	286,124
Different individuals ¹¹	31,729	26,324

4 Findings

To understand how knowledge in the form of questions and answers is created, shared, and curated, we first identified and categorized the main types of knowledge artifacts contained within messages on the R-help mailing list and Stack Overflow ‘R tag’ (RQ1). The emerging categories formed a typology and allowed us to identify and describe two approaches for constructing knowledge that are supported by these channels (RQ2). Interestingly, we found that some developers are active on both channels, and in some cases, even post the same questions. As a result, we investigated the benefits they gain by doing so (RQ3). We also present our findings about participation on the two channels over time (RQ4) and look closely at the different levels of participation activity between community users (RQ5).

¹⁰ <https://data.stackexchange.com/stackoverflow/query/new>

¹¹ Note: Stack Overflow data about user activity only goes up to September 2014.

4.1 What Types of Knowledge Artifacts Are Shared on Stack Overflow and the R-help Mailing List

To answer RQ1, we randomly sampled 400 threads of messages from both Stack Overflow and the R-help mailing list, where each thread included a question and the associated responses. We identified five main types of artifacts that capture knowledge: (1) Questions (2) Answers (3) Updates (4) Flags (5) Comments

Through our analysis, we further divided these types into sub-types—Table 3 presents our typology of knowledge artifacts, their descriptions, and their frequency in the sample of 400 threads in each channel. Even though we did not aim for a statistically significant sample size, the size of this sample (400 threads in each channel) guarantees a confidence level of approximately $95\% \pm 5\%$ for both channels. Using the Chi-square test of independence, we tested whether the distribution of types and sub-types of questions were different between the two channels. In all cases, they were found to be statistically different (with $p \ll 0.001$ in all cases).

Table 3: Typology of knowledge artifacts found on both Stack Overflow (SO) and the R-help (RH) mailing list and their frequency in the analyzed sample. Numbers in bold represent the most significant differences between the two sets.

		SO	RH	Prop SO	Prop RH
Questions					
<i>How-to</i>	Asks how to do something specific.	166	103	41.50%	25.75%
<i>Bug/Error-Exception</i>	Asks for a solution or reasons for an error message.	27	48	6.75%	12.00%
<i>Discrepancy</i>	Asks about an unexpected result of a specific function, process, or package.	53	88	13.25%	22.00%
<i>Set-up</i>	Asks for possible ways to set up the R environment before or after deployment.	15	31	3.75%	7.75%
<i>Decision help</i>	Asks for advice in making a decision.	36	35	9.00%	8.75%
<i>Conceptual-Guidance</i>	Asks for conceptual clarification or guidance on topics related to R or statistics.	48	49	12.00%	12.25%
<i>Code review</i>	Asks for a code review, explicitly or implicitly.	34	21	8.50%	5.25%
<i>Non-functional</i>	Asks for help (or suggestions) with a non-functional requirement such as performance or memory usage.	14	11	3.50%	2.75%
<i>Future reference</i>	Asks a question (often self-answering it) that might not exist on the channel, but that is interesting enough to warrant a thread for future reference.	5	4	1.25%	1.00%
<i>Other</i>	Asks for assistance unrelated to the channel, or the message contains unrelated information (e.g., announcements, ideas for improvement).	2	10	0.50%	2.50%
		400	400	100%	100%
Answers					
<i>Redirecting</i>	Provides a link to an existing solution that is not in the thread (e.g., external application, tutorial, project).	163	87	20.20%	15.03%
<i>Tutorial</i>	Provides a set of steps to teach people how to solve the issue.	105	15	13.01%	2.59%

<i>Source code</i>	Provides a source code snippet as the solution without an extensive explanation about the answer.	198	102	24.54%	17.62%
<i>Clue/Suggestion/ Hint</i>	Provides a possible way to fix the issue without actually solving it.	43	105	5.33%	18.13%
<i>Alternative</i>	Provides a different approach to a solution that is related to but not exactly what is being asked (e.g., mathematical approach, data structure modification).	33	98	4.09%	16.93%
<i>Explanation</i>	Provides an explanation of an approach that answers the question and lists steps on how to implement it.	203	101	25.15%	17.44%
<i>Announcement</i>	Provides a notification about some artifact (e.g., packages, libraries).	8	33	0.99%	5.70%
<i>Benchmark</i>	Provides a benchmark of multiple solutions posted by others or compares different answers.	5	3	0.62%	0.52%
<i>Opinion</i>	Provides an opinion or an expansion of another answer by including scenarios and examples.	49	35	6.07%	6.04%
		807	579	100%	100%
Updates					
<i>Announcement</i>	Announces specific events (e.g., bounties, future updates).	27	3	4.40%	1.21%
<i>Background</i>	Adds additional context to the question or answer.	74	57	12.07%	23.08%
<i>Correction</i>	Corrects format, grammar, spelling, and semantic mistakes.	301	2	49.10%	0.81%
<i>Expansion</i>	Expands the question or answer by providing scenarios or examples.	116	83	18.92%	33.60%
<i>Explanation</i>	Explains or clarifies a specific point in the question or answer, such as why the user chose a specific data structure, or the meaning of a variable.	83	95	13.54%	38.46%
<i>Solution</i>	The user answers their own question.	12	7	1.96%	2.83%
		613	247	100%	100%
Flags					
<i>Off-topic/ Opinion</i>	Identifies questions that are unrelated to the channels' interests, or requests answers based on opinion.	22	19	27.16%	35.19%
<i>Not an answer</i>	Indicates answers that are out of scope of the question or that do not answer the question.	0	27	0.00%	50.00%
<i>Repeated question</i>	Notifies a user that the question has been answered previously.	48	8	59.26%	14.81%
<i>Too localized</i>	Questions that are too specific and might not help future readers.	6	0	7.41%	0.00%
<i>Unclear</i>	Questions that are difficult to understand.	5	0	6.17%	0.00%
		81	54	100%	100%
Comments					
<i>Clarification</i>	Provides (or requests) additional information about a question or answer.	98	28	17.44%	10.49%
<i>Expansion</i>	Provides additional information.	127	65	22.60%	24.34%

<i>Correction/Alternative</i>	Suggests a change to a question or answer, offers an alternative solution or a correction.	102	89	18.15%	33.33%
<i>Compliment/Critic</i>	Posts something good, offers thanks, or provides an opinion or criticism.	157	52	27.94%	19.48%
<i>External reference</i>	References an external resource.	78	33	13.88%	12.36%
		562	267	100%	100%

Questions and Answers: Questions express one or more problems or concerns faced by a user on the R-help mailing list or on Stack Overflow, whereas answers represent solutions to questions. We observed that the types of questions on Stack Overflow are more specific than those on the R-help mailing list and are more likely to be tutorials. Stack Overflow also has more answers per question: 2 per question compared to 1.4 for R-help (see Table 3). However, R-help answers tend to offer more suggestions or alternatives than Stack Overflow answers.

Updates: An update is a modification of a question or answer. In Stack Overflow, updates are presented in one of two ways:

Labeled updates are explicitly shown in the body of questions or answers next to a label that identifies the update (e.g., edit, update, and p.s.). When multiple update labels appear in a message, each label is accompanied by a number (e.g., “[Edit 1:]”), a date (e.g., “Edit/Update (April 2011):”), or a bulleted list (e.g., “EDIT: -anova... -drop1...”).

Non-labeled updates are only visually recognizable through the message history system. The only indication of the change is a box at the end of the message that identifies the user who performed the change and the date when it occurred.

We found that non-labeled updates are often used to correct formatting, grammar, semantic mistakes and spelling, or to incorporate explanations, examples and suggestions without changing the meaning of the question or answer. Labeled updates are for everything else.

On the R-help mailing list, all communication occurs through email and authors do not explicitly tag messages as updates. For this reason, we define an update on R-help as *a message sent to a thread where the author has already participated once*.

Regarding update frequency in our sample, the Stack Overflow R tag contained 2.5 times more updates than the R-help mailing list. Corrections are more common on Stack Overflow (almost 50%), while R-help updates are often related to the adding of information to a thread (providing background, expansion, and explanation).

Flags: Flags are used to alert users that a question or answer does not match community expectations. Stack Overflow contains a flagging mechanism that’s often used to get a moderator’s attention. These flags can accomplish various objectives: mark a message as containing spam or rude/abusive behavior, or identify duplicate questions, off-topic messages, unclear questions, opinion-based questions, and low-quality answers. Depending on the type of flag, this can lead to a thread being closed or the loss of user reputation points.

The R-help mailing list doesn't have a built-in flagging mechanism. However, R-help users utilize the concept of flags, which we define as *messages used to call the attention of other community users*, similar to the way flags are used in Stack Overflow. In terms of their frequency, R tag posts on Stack Overflow contained 1.5 times more flags than posts on the R-help mailing list. Stack Overflow flags are primarily used to mark repeated questions. In contrast, flags on R-help are often used to indicate that a previous answer is incorrect.

Comments: In Stack Overflow, comments are considered “temporary ‘Post-It’ notes left on a question or answer”¹². Comments are located below each question or answer and can be used to clarify information or follow up with further details. On the R-help mailing list, we define comments as messages written to *improve an answer or as a follow-up to a discussion*. It should be noted that in order for an email to qualify as a comment, it should not be written by the person who asked or answered the original question (otherwise, the message would be considered an update). Because both Stack Overflow and the R-help mailing list permit participants to ask multiple questions in the same thread, the sub-categories of comments are not mutually exclusive.

Regarding the frequency of comments, the main difference between the two channels is that Stack Overflow comments are less likely to be considered corrections or alternatives (Correction/Alternative sub-category) than on the R-help mailing list. The Stack Overflow R tag sample also contained 2.1 times more comments than the R-help sample (see Table 3).

4.2 How Knowledge Is Constructed on Stack Overflow and the R-help Mailing List

Our analysis helped us identify two different approaches used for constructing knowledge (RQ2) on Stack Overflow and the R-help mailing list: participatory knowledge construction and crowd knowledge construction.

Participatory knowledge construction is an approach where answers are created through the cooperation of multiple users in the same thread. Participants complement each other's solutions by discussing the pros and cons of each answer and by adding different viewpoints, additional information, and examples. This process is similar to a team working together towards a common objective.

Crowd knowledge construction leverages the experiences of many users who work in a relatively independent manner. Each user contributes to the thread, adding variety to the pool of solutions. However, the user's priority is to provide a correct answer and not to discuss other solutions. This is comparable with the concept of a group in which people work towards the same objective but not necessarily together (e.g., Amazon's Mechanical Turk). Participants can vote on other's ideas, but the main idea is not constructed through discussion.

On the R-help mailing list, *participatory knowledge* construction occurs when: (1) previous answers are included in the current answer with clear links between them; or (2) a reply contains a direct reference to other answers or authors. Figure

¹² <http://stackoverflow.com/help/privileges/comment>

2 depicts two examples of the way participatory knowledge occurs on the R-help mailing list: direct citation of the author of a previous answer, and inferable links between answers.

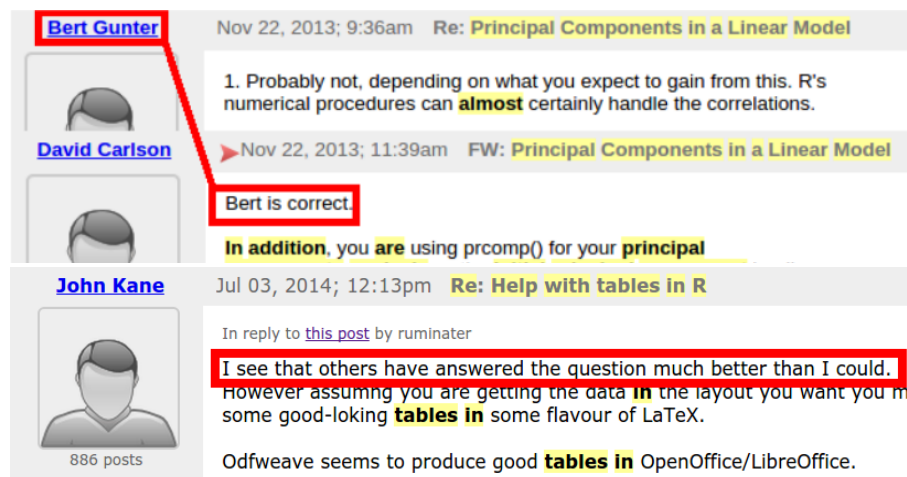


Fig. 2: Participatory knowledge construction on the R-help mailing list.

On Stack Overflow, *participatory knowledge* construction takes place when: (1) one can infer a link between answers, through either a direct or indirect reference; or (2) comments complement the answer or directly cite another author. Participatory knowledge construction also occurs in different places on Stack Overflow, perhaps as a consequence of its rich interface. We observe this type of knowledge construction when a user answers a question and directly cites or links to someone else's answer in the thread, or when a user cites someone else's question or answer in a comment (a typical case is linking to a previously asked question). Figure 3 depicts an example of participatory knowledge construction on Stack Overflow: when an answer was deemed insufficient, a user helped out by adding a comment and referencing another author's answer.

On the R-help mailing list, *crowd knowledge* construction takes place when different messages respond directly to the original question, rather than to another response.

On Stack Overflow, *crowd knowledge* construction occurs when: (1) there is no direct or inferable reference between answers; or (2) an answer is a variation of one of the other answers in the thread. Figure 4 depicts an example of crowd knowledge construction on Stack Overflow. As can be seen from the figure, two of the three answers provided the same solution.

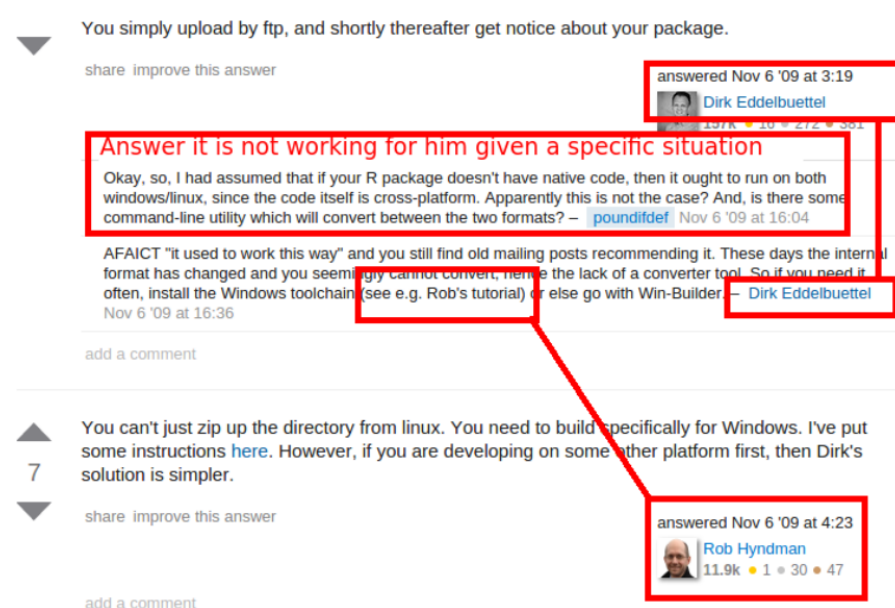


Fig. 3: Example of participatory knowledge on Stack Overflow. Users built on the comments and answers of other users.

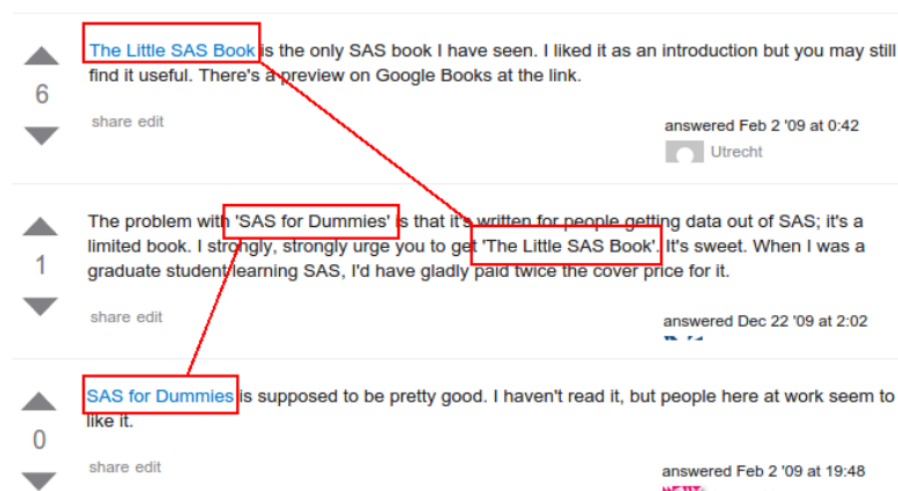


Fig. 4: Example of how crowd knowledge construction occurs. The three authors provided similar answers, but did it independently of each other.

4.3 Why Users Post to a Particular Channel or to Both Channels

From our survey, we were able to learn why some R community users preferred one channel over the other; we summarize their responses below. Using results from the analysis of the archived data, we then present why some users post some questions to both channels.

4.3.1 *Why participants post on Stack Overflow*

Survey participants preferred using Stack Overflow for several reasons: (a) the ability to gain peer recognition (the advantage of gaining points—and visibility—is a major draw of Stack Overflow); (b) its rich and user-friendly interface; (c) answers are straight to the point; (d) questions are usually answered faster on Stack Overflow than on the R-help mailing list; and (e) it is easy to search for previous questions and answers.

However, the respondents reported a few main drawbacks of using Stack Overflow: (a) there is an overabundance of related questions; (b) one requires a certain level of experience to understand some of the answers; and (c) Stack Overflow's strict rules only allow questions and their answers, they do not allow discussions nor questions about opinions.

4.3.2 *Why participants post on the R-help mailing list*

Survey participants reported a few benefits of using the R-help mailing list: (a) the email format is convenient; (b) following the mailing list provides awareness and increases learning in new topics; (c) there is more flexibility regarding the topics that one can discuss; and (d) there is a high level of participation from experienced users. The respondents did note a couple of disadvantages of R-help: (a) some discussions lead to aggressive behavior; and (b) searching the archives is not easy.

4.3.3 *Why participants post to both channels*

Our analysis of the archived data revealed that some users (79 cases in our sample) posted the same question on both channels. Based on the responses from the survey, we identified that being active on both channels brings benefits to those asking and answering questions (RQ3):

Find a better answer: As expected, two channels are better than one— one channel might result in a better answer than the other.

Support follow-up questions: We found that the R-help mailing list is often used to conduct follow-up discussions on specific answers provided to Stack Overflow questions. Stack Overflow's focus is on finding an answer to a question and does not provide an environment to discuss the specifics of an answer (unless it is asked as another question). In contrast, a discussion on R-help can continue long after an answer has been found through follow-up questions, and not only by the person who asked the original question.

Speed up answers: Users ask the same question on both channels in order to get an answer faster. However, this behavior is not encouraged by the community as it is deemed impolite¹³.

¹³ <https://goo.gl/p9vVaj>

4.4 How Participation Differs Between the Two Channels Over Time

To bring insights to RQ4, we first discuss how user participation differs between the two channels over time and then consider how long users participate.

4.4.1 How user participation differs between the two channels over time

Vasilescu *et al.* studied the participation of the R-help and Stack Overflow communities over time. Their research showed strong evidence that knowledge seeking activities were moving from R-help to Stack Overflow (as of the end of 2013) [23]. These findings prompted us to also consider how participation differs over both channels over time (RQ4).

We first explore the evolution of the number of questions in both channels as the main proxy of activity (the results are shown in Figure 5). As reported in [23], the trend indicates that Stack Overflow continues to grow and R-help continues to decrease. One aspect that is changing in Stack Overflow is the growth of questions that receive an overall negative score. A score in Stack Overflow is the sum of positive votes minus negative votes, which can be considered a proxy of the question's quality. As can be seen, the number of questions with an overall positive score has flattened and is starting to decrease. However, looking at the trends over the last 10 years, this might be misleading. Figure 6 shows only the number of questions since Jan 2015. As can be seen, both channels are relatively flat in terms of overall questions. The number of questions in Stack Overflow is between 10 and 20 times the number of questions in R-help, but the number of questions with a positive overall score in Stack Overflow is starting to decrease (we recognize this might be a temporary effect).

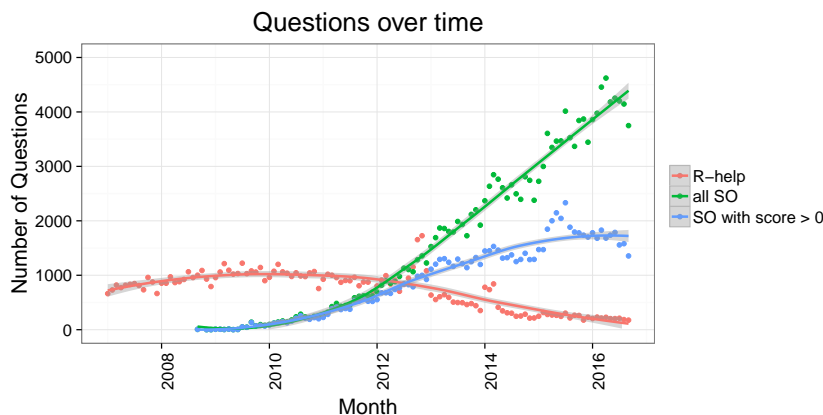


Fig. 5: Number of questions asked over time. As can be seen, Stack Overflow activity has been much larger than R-help activity. However, the number of questions with positive score has flattened.

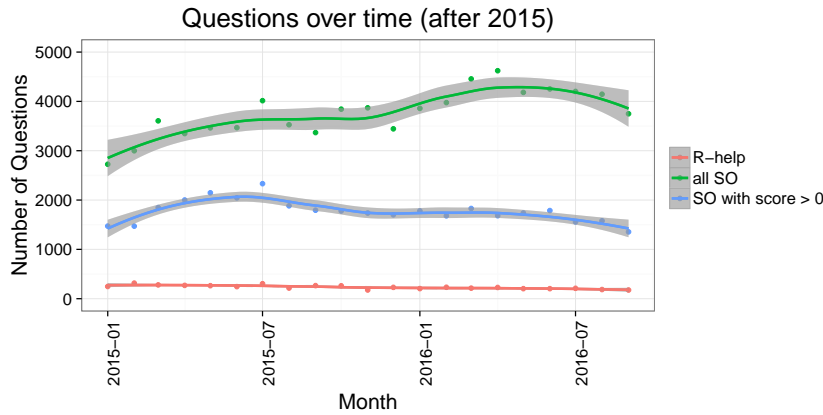


Fig. 6: Detail of Figure 5 showing only activity after January 2015. As can be seen, both channels have flattened, but the number of Stack Overflow questions with a positive score is starting to decrease.

4.4.2 How long users participate

Another important measure of the community and its ability to curate and maintain knowledge is whether their users continuously participate over time, even if their participation is small. To measure the continuity of participation, we divided the history of both channels into one month segments. In R-help, we considered that a user participated in a given month if they posted at least one email to the list during that period. For Stack Overflow, we considered that a user participated in a given month if they posted, responded to, or commented on at least one question.

The results show that users do not participate in either channel for a long time. Figure 7 shows the accumulated proportion of users that participated during a given number of months (not necessarily consecutively). The curves are very similar and skewed: 62% of R-help users and 65% of Stack Overflow users are active for a single month; 90% of R-help users are active for 5 months or less; 90% of Stack Overflow users participate 4 months or less.

4.5 Are There Significant Differences in Participation Activity between Community Members?

To consider participation activity in the two channels, we first consider new users. We explore how many newcomers there are to both channels and consider if they are likely to post again. Lastly, we look at contributors that are responsible for the vast majority of posts in both channels.

4.5.1 Activity of new users

New users are important for the survival and continuity of a community. Unfortunately, since we can only track users when they actively participate in a channel, we

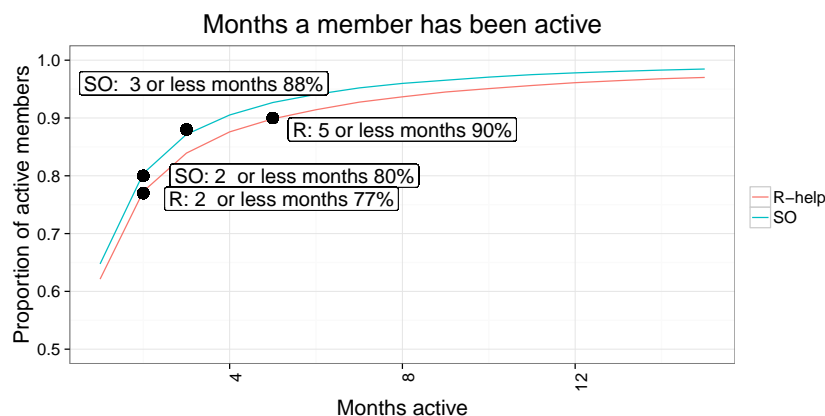


Fig. 7: Months a user has been active. This plot shows the accumulated proportion of users who have been active for a given number of months. As can be seen, 62% of R-help users and 65% of Stack Overflow users are active for 1 month only; 90% of R-help users are active 5 or months or less; 90% of Stack Overflow users participate 4 or months or less. Months do not have to be consecutive.

do not know who is a passive participant. Hence we use the date of a first contribution as the proxy of when a user joins the community. Figure 8 shows the proportion of users who have posted their first question in a particular month. As can be seen, both communities show a similar pattern: between 25% and 35% of the users who post new questions are new during that month. However, we found that many of these users participate only once: they post an initial question and then never post again (question or answer, see Fig. 9). One year after the creation of Stack Overflow, around 1 in 10 users only ever post a single question, while in R-help, the number has been increasing over time. There has been a visible decrease in the proportion of new users contributing to both channels. In fact, the number of users who only participate once (at any time) is relatively large over the lifetime of the channels: 43% in R-help and 32% in Stack Overflow.

4.5.2 Common users of both channels

The number of common users we identified across both datasets is relatively small (around 2.5% of all users). Yet a handful of these contributors are responsible for a very large proportion of the answers in both channels. Figure 10 shows the accumulated proportion of answers that have been contributed by these authors. As can be seen, the top contributor has contributed 3.9% and 3.7% of answers in R-help and Stack Overflow, respectively. In fact, the top 6 contributors to both channels are responsible for 10% of the answers (and only one of those belongs to the *R-core* group). Here, answers on Stack Overflow include both posts and comments to questions.

4.5.3 Prolific contributors

Both channels benefit from a handful of very prolific members (also referred to as “caretakers” by Srba and Bielikova [18]) who answer most questions. Thus, we re-

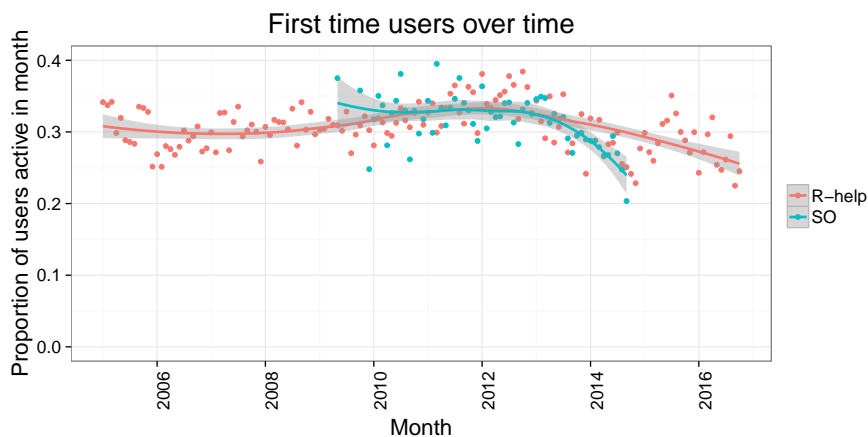


Fig. 8: Proportion of new users over time. The top lines (thicker) correspond to the number of users who post their first question in that month. The bottom (thinner) lines correspond to the subset of new users who only ask one questions and then never participate again.

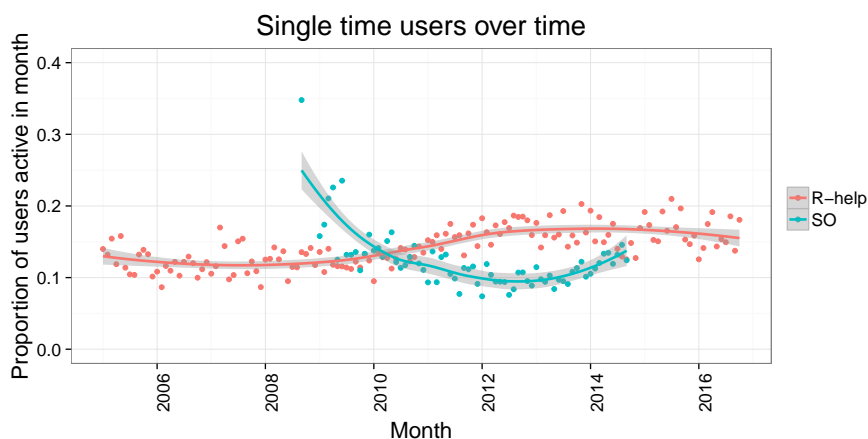


Fig. 9: Proportion of one time users in any given month.

fer to these users as *prolific contributors*. Figure 11 shows the prolific contributors in each channel, ordered by the number of answers they have contributed. The vertical axis shows the accumulated proportion of contributions to all answers in each channel. As can be seen in R-help, 9 users have contributed 25% of all answers and 49 have contributed 50% of all answers. In Stack Overflow, the distribution is not as steep, yet 24 users have contributed 25% of all answers and 131 have contributed 50% of all answers (including comments). This means that 0.15% of the R-help mailing list users contribute 50% of all the answers, while 0.45% of the Stack Overflow users contribute 50% of all the answers. To put this into context, the prolific contributors in R-help post between 2 and 3 answers a day on average, while the prolific contributors of Stack Overflow average between 3 and 5 answers a day.

Relative accumulated contribution of common contributors to both channels

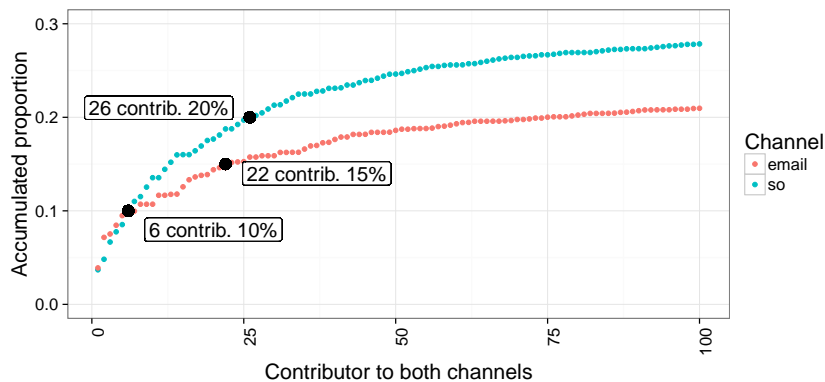


Fig. 10: Accumulative proportion of contributions by the most prolific users who post to both channels. As can be seen, the top 6 most prolific users contribute approximately 10% of the answers.

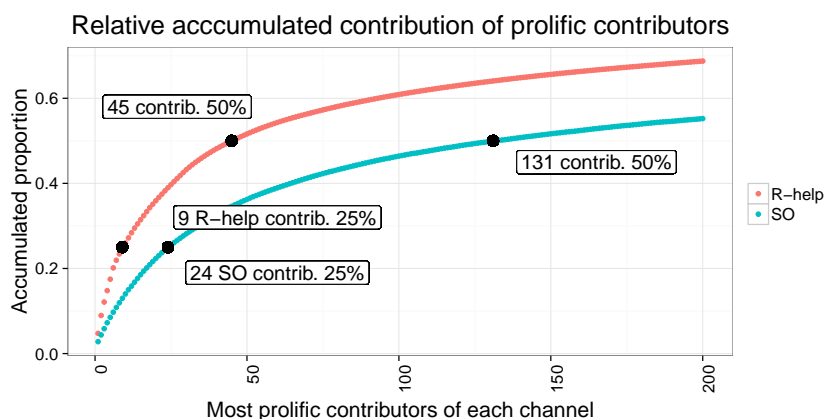


Fig. 11: Accumulative proportion of contributions by the most prolific users of each channel. As can be seen, the top 9 and top 24 contributors to R-help and Stack Overflow, respectively, are responsible for 25% of all the answers.

5 Discussion

In this section, we reflect on the results presented in the previous section, distilling some themes from this work, relating them to relevant related research and pointing out possibilities for future work. We then derive and share recommendations that members of a community may follow when they have the option of posting questions (and answers) to multiple Q&A channels. As we present these themes and recommendations, we provide representative quotes extracted from the survey we conducted in the second phase of our study, using P# to indicate the participant ID.

5.1 Knowledge Creation and Curation

Based on our results, we found that both channels provide similar knowledge support for asking questions and providing answers. However, there are some important differences between the channels we studied which we discuss in detail below (and summarize in Table 4).

Table 4: Comparison of the ways knowledge is shared on Stack Overflow and the R-help mailing list.

	Stack Overflow	R-help
Knowledge construction	Mainly crowd	Mainly participatory
Topic restriction	Yes	No
Emphasis	Curating knowledge	Developing knowledge

5.1.1 Knowledge construction

Stack Overflow’s gamification mechanism encourages many users to respond first when answering questions [16]. In contrast, the R-help mailing list is a less competitive environment where users tend to build on other responses. On R-help, users work as a team rather than acting as individuals searching for points (as is the case on Stack Overflow). As a result, knowledge on Stack Overflow is built in a more crowdsourced manner, while knowledge on the R-help mailing list is usually built in a participatory manner.

Since, the competitive Stack Overflow environment creates an incentive to be the first to answer rather than improve and build on other answers, it is common to find a question with several answers that provide the same information. For example, three of the six answers in the Stack Overflow question titled “*Resources for learning SAS if you are already familiar with R*”¹⁴ referenced the same books. And while Stack Overflow provides a powerful curation mechanism to ensure the best answers make it to the top, this mechanism does not explain why an answer is better than another.

In contrast, the R-help mailing list tends to be more participatory in how users construct knowledge. It fosters an environment where users discuss proposed answers—users tend to provide more background to answers and explain the rationale behind them. For example, the question “*Arrange elements on a matrix according to row-Sums + short ‘apply’ Q*” was posted to both Stack Overflow¹⁵ and R-help¹⁶. This question illustrates the contrast in how the two communities build knowledge. On Stack Overflow, each participant contributed a solution without any evidence of collaboration with others. Whereas users on the R-help mailing list complemented each other’s answers by providing further information and insights to existing answers. Vasilescu *et al.* [23] showed that users who are active in both channels tend to provide answers faster on Stack Overflow than on R-help, suggesting that they are motivated

¹⁴ <http://goo.gl/Mb4Pbk>

¹⁵ <http://goo.gl/a8AES8>

¹⁶ <http://goo.gl/PGf1T5>

by the gamification aspects of Stack Overflow, and thus tend to gravitate towards crowd knowledge construction.

While the crowd-based approach is prevalent, the construction of knowledge on Stack Overflow is not limited to the crowd-based approach. Participatory knowledge construction can also be observed, such as the up/down voting of questions and through the provision of comments. However, in most cases, participatory knowledge construction on Stack Overflow is used for editing answers (e.g., correcting grammar) or linking to previously asked questions. Similarly, some knowledge on the R-help mailing list is constructed in a crowd-based manner, but this is much less prevalent than participatory construction.

Tausczik *et al.* [21] examined how users of Math Overflow (a Q&A platform for mathematicians) collaborate and construct knowledge. They found that collaboration was diverse and fell on the spectrum between *independent* (crowd-based) and *interdependent* (participatory). Similar to our findings with Stack Overflow, the most common collaborative act was of an independent nature (i.e., providing information), while other contributions that built on existing work were less common (i.e., clarifying the question, critiquing answers, revising answers, and extending answers).

Our results seem to imply that Stack Overflow's gamification features, while highly effective, have the side effect of reducing collaborative knowledge creation between users. In their study on building Stack Overflow reputation, Bosu *et al.* [4] proposed six strategies for increasing reputation score, two of which were *to be the first to answer*, and *to answer at off-peak hours*, indicating crowd knowledge creation. Furthermore, while Stack Overflow gives people the ability to vote on comments, it does not reward points to users that post comments. For example, some users search Stack Overflow for answers within comments and convert them to proper answers to gain reputation points¹⁷.

An important research question that arises from these findings is whether Stack Overflow's model can be improved to provide better participatory knowledge construction support without hindering its ability to curate information for future use.

5.1.2 Topic restriction

Stack Overflow's participation rules only permit questions that have a clear answer, restricting topics discussed. In contrast, the R-help mailing list is suitable for discussing any topic related to the R language. For example, questions related to R but not focused on software development are not rejected by the R-help mailing list community—topics that trigger discussion are welcomed.

Stack Overflow questions that trigger a discussion are flagged as opinion-based or off-topic and typically closed. Correa and Sureka [5] found that 18% of deleted questions on Stack Overflow are subjective (i.e., ask for opinion). For example, a question titled "*What's a good example of really clean and clear [R] code, for pedagogical purposes?*"¹⁸ was flagged as *off-topic* because the question was not related

¹⁷ <http://duncanlock.net/blog/2013/06/14/the-smart-guide-to-stack-overflow-zero-to-hero>

¹⁸ <http://goo.gl/9JjZW1>

to software development. An R-help user wrote a fine explanation of the purpose of each channel in a message on the mailing list:¹⁹

“Got an R programming question that you think has a definite answer? Post to [Stack Overflow]. Want to ask something for discussion, like what options there are for doing XYZ in R, or why `lm()` is faster than `glm()`, or why are these two numbers not equal? Post to R-help. Questions like that do get posted to [Stack Overflow], but we [vote] them down for being off topic and they disappear pretty quickly.”

Squire reported that, despite the gains in participation and the response time provided by Stack Overflow, many development communities keep using mailing lists, either as a primary communication channel or as part of a hybrid solution where multiple channels are used, thus allowing for non-restrictive topics and fostering of discussion [17]. Mailing lists are also favoured for their simplicity, and for allowing guaranteed delivery (i.e., knowing who will receive the email) [25].

5.1.3 Curated knowledge and knowledge development

One of the main benefits enabled by Stack Overflow’s crowd-based knowledge construction is the creation and curation of a pool of questions and answers. In contrast, R-help provides an environment in which users develop knowledge through participation, but this knowledge is not curated for future use. This makes the information difficult to be reused by those who were not participants (either active or passive) during its creation.

While Stack Overflow has been successful, some users feel that by not fostering discussion, it restricts thinking that might lead to better answers, as P26 explained:

“Many developers share my view that [Stack Overflow] is a very bad model, ... [it] removes the value added by reading list traffic that doesn’t seem directly relevant to a currently conceptualized question, but which may lead to a new conceptualization (out-of-the-frame thinking). [Stack Overflow] cannot do that.”

Similarly, P35 stated that they use the R-help mailing list if the questions are not 100% “help-me-to-code-this”.

However, Stack Overflow shines when questions have to be kept for posterity. Its curation mechanisms provide tools for keeping the channel clean of what seems to be unnecessary information (e.g., flagging questions, deleting comments, editing messages, and demoting irrelevant answers), as P14 explained:

“[Stack Overflow] is an excellent model for providing a rich resource for users of R, which the R-help mailing list was not. Ability to include light markup, render code blocks nicely, no nested email threads all helps the experience of searching for and finding the help that a user needs, and I want to contribute to that.”

¹⁹ <http://goo.gl/mTccwx>

Another interesting aspect emerging from our findings is that the activity on the R-help mailing list is only marginally smaller than on Stack Overflow (the proportion of responses in each category fluctuated between 1.4 and 2 times). Further research is required to assess and verify the quality and effectiveness of answers.

5.1.4 Maturing as a community: from knowledge creation to knowledge curation

One of the major discoveries from Phase III of our study is the slowing of the growth of active participation in Stack Overflow (asking or answering questions). In particular, the number of new questions with an overall positive score is starting to decrease over time. This is to be expected: most sought after questions have probably been asked already. Given the very large number of questions already asked through Stack Overflow, many questions today are expected to be either duplicates of previous questions, or to be questions of low intrinsic value (too specific to be useful to others), or to end up being flagged as not a question. This decline of participation on Stack Overflow has been discussed by community members²⁰ and the Stack Overflow development team²¹, both agree that Stack Overflow “*is not declining, it is serving its purpose quite well*”.

When we consider that the popularity of R continues to increase, this implies that when new members to the R community seek answers they might not need to post questions—a testament to the value of the knowledge currently gathered by Stack Overflow. This also seems to imply that the activities of Stack Overflow contributors are shifting from answering new question to tagging, flagging (e.g. as duplicates, unclear, or off-topic), or editing existing questions.

The frequency of postings on the R-help mailing list has now stabilized on approximately 200 questions per month. It is possible that Stack Overflow, by curating answers to most frequently asked questions, has contributed to a reduction of questions posted on the R-help mailing list, and the R-help community may now be able to concentrate on higher-level questions than those asked in the past. Further research should verify this hypothesis.

5.1.5 Active vs. Passive participation

We found that approximately 43% of participating members on the R-help mailing list only ask a question once (and never actively participate again, neither asking nor answering any additional questions). In Stack Overflow, the proportion is 33%. It is not clear why a person participates only once, perhaps these persons continue to passively participate (e.g., consuming content, voting up/down).

Similarly, approximately two thirds of users in both channels only participate for a short period of one month and never actively participate again. This might indicate that some users are sporadic users of R—perhaps learning it— and use it only for a short period. But more research is necessary to know the true reasons. We have observed in

²⁰ <https://www.javacodegeeks.com/2016/09/stopped-contributing-stackoverflow-not-declining.html>

²¹ <https://stackoverflow.blog/2016/10/podcast-89-the-decline-of-stack-overflow-has-been-greatly-exaggerated/>

the set of common users to both channels that some of them migrate from one channel to the other. But the majority simply stop participating. It is very likely that they continue to use Stack Overflow as passive users; hence, benefiting from the channel but without contributing back. Further research is needed to characterize both active and passive participation patterns.

Nonetheless, new users appear to be joining both channels all the time. Between 25% and 35% of users in any given month are new.

5.1.6 Prolific contributors

There is a handful of users that are responsible for a large proportion of answers. In R-help, we found that 50% of the answers have been contributed by 0.15% of the users. While in Stack Overflow, 0.45% of the users have contributed 50% of the answers. It is not clear if the success of both channels can be attributed to them or not. It is possible, for example, that without them, other users might have answered those questions. Further research should look into the motivations of these individuals, consider the time they commit to help others, and also compare these patterns of activity with similar channels to see if this is a common phenomenon.

Several of these prolific contributors actively participate in both channels. The top one is responsible for almost 4% of the answers in either channel. Combined, the top 6 contributors have authored approximately 10% of the answers of either channel. These prolific contributors are very likely serving as a bridge of knowledge between the channels, moving knowledge from one channel to the other. This should be confirmed in future work.

5.2 Recommendations for Using Multiple Q&A Communication channels

One of the expected outcomes of this study is a set of recommendations for using multiple communication channels. Other research on Stack Overflow also points to these and other recommendations. For example, Yao *et al.* discuss the importance of how the phrasing of a question influences the quality of the answer provided [24], and Anderson *et al.* emphasize the importance of badges (i.e., gamification) on participation [1,2]. In the following, we build on this existing literature as well as our results to provide five recommendations for people seeking or contributing answers in community forums. Our recommendations are summarized in Table 5.

Table 5: Recommendations to improve the benefits experienced from using several Q&A channels.

Choose the correct channel
Be aware of the channel rules and the basic concepts and nomenclature used
Provide good background to the question
Learn to use external resources
Behave altruistically

5.2.1 Choose the correct channel

Each channel provides a list of *topics* that are deemed acceptable. The topics are regulated either by the community or the channel’s moderators. “...*Stack Overflow* has (a) more limited range of help topics (help for code only), whereas *R-help* is broader (philosophy, posting announcements, etc.)” [P35]. Knowing which channel is more suitable for a specific topic can improve the response time or the quality of the answer.

In some cases, it is expected that questions will be answered by a *specific group* (e.g., R-core team) regardless of the topic, as P32 stated: “*If I really want an answer from someone in R-core or closely related people, I would definitely choose the mailing list.*” For example, in the R-help thread “*Co-integration and ECM in Package {urca}*”²² a participant asked the R-core team how to solve a problem:

“Dear R Core Team, I am using package {urca} to do co-integration and estimate ECM model, but I have the following two problems...”

In this scenario, Websites associated with a specific package or library might be the best way to communicate directly with the creators of that technology. Thus, the R-help mailing list is a place for discussion and Stack Overflow is a place for questions that have a clear answer.

5.2.2 Be aware of the channel rules and the basic concepts and nomenclature used

Throughout this study, we noticed that most of the harsh responses were given to users who did not learn the participation rules or had not learned the basics of R or statistics (and were seen as wasting other’s time). For anybody using a channel, the community expects users to familiarize themselves with the channel in advance and learn the basics of the technology that they are discussing.

Stack Overflow provides user guides²³ for each of the main features of the channel, such as badges, questions, answers, flags, comments, and the reputation system. The R-help mailing list only has general instructions²⁴ and a guide about posting on the channel²⁵.

We also discovered that the R community has developed resources to improve the quality of participation on the communication channels. For example, the post on Stack Overflow “*How to make a great R reproducible example?*”²⁶ provides tips and tricks for creating a reproducible example using the R language. Another example is the document “*How to write a reproducible example*”²⁷ which provides tips for posting a reproducible R code example to mailing lists: “...*Before putting all of your code in an email, consider putting it on <http://gist.github.com/>[GitHub*

²² <http://goo.gl/7olLv7>

²³ <http://stackoverflow.com/help>

²⁴ <https://www.r-project.org/mail.html#instructions>

²⁵ <https://www.r-project.org/posting-guide.html>

²⁶ <http://stackoverflow.com/questions/5963269/how-to-make-a-great-r-reproducible-example>

²⁷ <http://adv-r.had.co.nz/Reproducibility.html>

Gist app]. It will give your code nice syntax highlighting, and you don't have to worry about anything getting mangled by the email system..."

Finally, there are manuals like “*An Introduction to R*”, and the FAQ Web pages for R that are available to the public—most of the time, free of charge—and from which any user can learn the basics of R. For example, the R community provides a compendium of PDF documents for new users of different languages.²⁸ In Stack Overflow, supported technologies are provisioned with Web pages and links to free and paid materials.²⁹

5.2.3 Provide good background to the question

In spite of reading the documentation, a user may fail to use the channel appropriately. The community may feel that the question asked, the information provided, or something else entirely is not in compliance with the expectations and rules of the channel. In such cases, one should describe the documentation they read before posting, the attempts made to answer a question if they asked one, and the goal(s) they want to achieve in the question or answer posted. For poor questions, this would avoid the user having to receive answers like “*read the manual*” or “*read the posting guide*”. For example, in the thread “*lme4 GLMM*”³⁰ the user explicitly acknowledged the repeated question and explained the rationale for doing so: “*I'm very sorry for my repeated question, which I asked 2 weeks ago, namely: I'm interested in possibly simple random-part specification in the call...*”

5.2.4 Learn to use external resources

A common practice when answering or asking questions is to provide links to documentation, examples, source code, or other resources. As links point to online resources that may no longer be available in the future, it is important to include the key points of the resource within the question or answer. For instance, when a question or answer contains information in an external file hosting service like Dropbox or Google Drive, the owner of the service account can remove or break the link at any moment, leaving the message incomplete or impossible to reproduce. P33 suggested that “*Questions should be self-contained as much as possible. Exceptions: recognizable links such as CRAN, R documentation, etc.*”

Based on our observations, we provide the following set of recommendations for using third-party resources within links.

Use well-maintained Websites that are expected to be available in the future, such as Wikipedia and the official documentation in CRAN. For example, a user on Stack Overflow posted: “*I'm doing a simulation where I need to calculate a [Wikipedia link] convolution of [Wikipedia link] multinomial distributions...*”

Use resources that support or expand the message to further clarify the message for those who might need it. For instance, a thread on Stack Overflow titled “*How do I*

²⁸ The R manuals are available at <https://cran.r-project.org/>

²⁹ Materials available at <http://stackoverflow.com/tags/r/info>

³⁰ <https://goo.gl/Gbek3R>

save all the draws from a MCMC posterior distribution to a file in R” states “...You should be able to open a text connection using `?file` [more information] with the open argument set to write...”

Use links rather than large attachments as it is not always practical to include all the related information in a message. Providing links to videos or large documents is usually preferred over including them as attachments. For example, in the R-help thread “Using *FUNCTION* to create usable objects”, a user linked to a PDF rather than quoting it: “*I suspect you are trying to find your way into Circle 6 of ‘The R Inferno’ but haven’t yet got in. [link to PDF of R Inferno].*”

5.2.5 Behave altruistically

It is obvious that users help others by answering questions. However, while analyzing questions and answers, we identified positive user behaviours that we believe are worth mentioning. These behaviours provide evidence of an altruistic way of thinking and the strong commitment that users have towards building knowledge in their community.

I answered my own question: Some questions are answered by the user that asked the question. They posted back to the channel to document their solution and help others: “*Just for the records (and if anyone ever wants to find the ‘solution’), I solved my own problem.*”³¹

I did it for you: When answering, authors provide source code to help others: “*I coded up the algorithm from the Cameron and Turner paper. Dunno if it gives exactly the same results as my (Splus?) code from lo [sic] these many years ago...*”³².

Updated or continued years later: Some questions are answered months or years later. For example, a user on Stack Overflow modified an answer to provide a more updated version of the source code³³, and a question asked on the R-help mailing list in 2012 was continued two years later.³⁴

Ideas to improve the channel: This behaviour is specific to the R-help mailing list. Sometimes users suggest modifications or new features to improve the channel. For example, a user proposed to create a package repository that can be accessible through a public wiki or version control interface.³⁵

5.3 Threats to Validity

Here we examine and discuss threats to the validity of our approach [15].

Construct validity: To reach the emerging themes, we relied on subjective human judgment during the data coding phase. Researchers had to decide if a message

³¹ <https://goo.gl/r3z0DX>

³² <http://goo.gl/GXWGG3>

³³ <http://goo.gl/k6ZARR>

³⁴ <http://goo.gl/kgSHZv>

³⁵ <http://goo.gl/p0IunD>

fell within a specific coding category. To alleviate this issue, two researchers coded the qualitative data as part of the analysis process. We applied the Cohen Kappa coefficient on categories that were not mutually exclusive, but whose purpose was to trigger discussion between coders. We set a threshold of 0.8 as the minimum to obtain agreeable results, which is higher than the 0.6 suggested in the literature [11]. Regarding the quantitative analysis. The main thread to validity is that in R-help, we consider all emails that start a thread as questions, and their follow-ups as responses. It is likely that a small proportion of emails are announcements or non-question emails. Stack Overflow stopped publishing the md5 of users, and hence it is likely that the number of common contributors between the two channels is underestimated.

Internal validity: Stack Overflow's data is structured while the R-help mailing list consists of unstructured data. As a result, some of the mapping between the two channels was straightforward (e.g., a follow-up to a reply is a comment to that question), while in other cases it wasn't as obvious (e.g., identifying some emails as questions). To reduce the risk of bias when mapping the messages between channels, two researchers performed the mapping.

External validity: Our case study was exploratory in nature and we purposefully aimed to study the R community. Many R users are likely to be *casual developers* with limited or non-existent programming experience, with backgrounds that vary from biology to statistics. Thus our findings may not apply to other developer communities. However, since Stack Overflow and mailing lists are widely used by other communities, we believe that our findings may be extended to these communities as well [17]. We do not claim the generalizability of our findings to other communication channels (e.g., Slack, GitHub), and further research is required to examine how knowledge is shared on other channels used by developers.

6 Conclusions

The purpose of this study was to understand how the R community collaborates when using different communication channels in the creation and curation of knowledge. In particular, we concentrated on studying how this community has used Stack Overflow (using the R tag) and the R-help mailing list to both ask and answer questions, through a random sample of 400 threads from each channel. Our research shows that both channels are active communication channels where participants are willing to help others.

We found that knowledge contributed in response to a question can be classified into four main categories: answers, updates, flags, and comments. The number of responses sent in each of these categories was between 1.4 and 2.5 times greater on Stack Overflow than on the R-help mailing list. While all four types of contributions exist in both channels, they exhibit differences. For example, on Stack Overflow, answers are more focused towards step-by-step tutorials, while R-help answers are more likely to be suggestions or alternatives. Similarly, on Stack Overflow, updates are focused on language (grammar and spelling), while on R-help, the updates are expansions on previous responses.

The analysis of these questions and answers shows that knowledge is constructed in each channel in a different manner. On Stack Overflow, there is a tendency to use a crowd approach: participants contribute knowledge independently of each other rather than improve other answers. This is likely a result of the gamification of Stack Overflow where the person who provides the best answer is the one that gains the most points. In contrast, the R-help mailing list uses a participatory approach where participants are more likely to build on other answers, collaborating towards finding the best solution.

Another important difference between both channels is that Stack Overflow focuses on making knowledge available for future retrieval. On the other hand, knowledge on the R-help mailing list focuses on the discussion of knowledge, but not in its long-term storage or retrieval. Respondents to our survey commented that while it is easy to find answers on Stack Overflow and make sense of them, on R-help it is not only hard to find the relevant answers to a question, but it is also hard to see how the many responses to a question relate to each other, and ultimately, what the best answer to the question may be.

Another result of our research is that we found that participants prefer Stack Overflow to ask questions that are expected to have a direct answer. They prefer to use the R-help mailing list when the question requests opinions (Stack Overflow forbids them) or when they expect to reach core developers of the R project. Some participants ask the same question in both channels in the hopes of gaining the advantages of both channels. Additionally, R-help has the ability to complement Stack Overflow by providing a medium where the rationale of answers can be discussed.

A major result of our quantitative study is the impact of a small fraction of the members of both channels who answer a large proportion of the questions in both channels. There is also a dedicated core of members who contribute to both channels and serve as avenues connecting the two communities. At the same time, a large proportion of members participate very few times over a very short period of time.

It was also interesting to observe that the frequency of new questions in Stack Overflow is starting to slow down, specially those with a positive score. This effect is probably due to the fact the most-frequently-asked questions in R have been already asked—an answered; the new questions are either inherently more difficult, or more specialized and therefore have a smaller pool of members who would benefit from them; however research is needed to understand this effect.

Overall, this research shows that the R community is committed to using both channels to help others. Each channel has advantages and disadvantages, and the community appears to be using both effectively to create and curate knowledge regarding the R language. We provided **recommendations** for community users that need to use these or other Q&A channels. Furthermore, our **typology of knowledge artifacts** that we summarized in Table 3 can be used by other researchers that wish to study and understand how knowledge is constructed and curated in other channels or across other communities. As new channels (such as Slack) become more widely adopted, studying these newer channels and comparing them to existing channels is an imperative aspect of understanding knowledge formation in software development.

7 Acknowledgments

The authors would like to thank Cassandra Petrachenko for the editing support and valuable comments that contributed to this work. We also thank Lorena Castañeda for her assistance with the data collection and analysis processes. Finally, we thank the R community users that responded to our survey.

References

1. A. Anderson, D. Huttenlocher, J. Kleinberg, and J. Leskovec. Steering user behavior with badges. In *Proc. of the 22nd Intl. Conf. on World Wide Web*, pages 95–106, 2013.
2. J. Antin and E. F. Churchill. Badges in social media: A social psychological perspective. In *CHI 2011 Gamification Workshop Proc.*, 2011.
3. N. Bettenburg, E. Shihab, and A. E. Hassan. An empirical study on the risks of using off-the-shelf techniques for processing mailing list data. In *ICSM'09: Proc. of the 25th Intl. Conf. on Software Maintenance*, pages 539–542, 2009.
4. A. Bosu, C. S. Corley, D. Heaton, D. Chatterji, J. C. Carver, and N. A. Kraft. Building reputation in stackoverflow: An empirical investigation. In *Proc. of the 10th Intl. Conf. on Mining Software Repositories, MSR '13*, pages 89–92, 2013.
5. D. Correa and A. Sureka. Chaff from the wheat: Characterization and modeling of deleted questions on stack overflow. In *Proc. of the 23rd Intl. Conf. on World Wide Web, WWW '14*, pages 631–642, 2014.
6. J. Creswell. *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches*. SAGE Publications, 2009.
7. D. German, B. Adams, and A. Hassan. The evolution of the r software ecosystem. In *Software Maintenance and Reengineering (CSMR), 2013 17th European Conf. on*, pages 243–252, March 2013.
8. C. Gomez, B. Cleary, and L. Singer. A study of innovation diffusion through link sharing on stack overflow. In *Proc. of the 10th Intl. Conf. on Mining Software Repositories*, pages 81–84, May 2013.
9. R. Ihaka and R. Gentleman. R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, 5(3):299–314, 1996.
10. H. Jenkins. *Confronting the Challenges of Participatory Culture: Media Education for the 21st Century*. The John D. and Catherine T. MacArthur Foundation Reports on Digital Media and Learning. MIT Press, 2009.
11. J. R. Landis and G. G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174, 1977.
12. H. Li, Z. Xing, X. Peng, and W. Zhao. What help do developers seek, when and how? In *Reverse Engineering (WCRE), 2013 20th Working Conference on Reverse Engineering*, pages 142–151. IEEE, 2013.
13. L. Mamykina, B. Manoim, M. Mittal, G. Hripcsak, and B. Hartmann. Design lessons from the fastest Q&A site in the west. In *Proc. of the SIGCHI Conf. on Human Factors in Computing Systems, CHI '11*, pages 2857–2866, 2011.
14. P. Naur. Programming as theory building. *Microprocessing and microprogramming*, 15(5):253–261, 1985.
15. P. Runeson, M. Host, A. Rainer, and B. Regnell. *Case Study Research in Software Engineering: Guidelines and Examples*. Wiley, 2012.
16. L. Singer, F. Figueira Filho, B. Cleary, C. Treude, M.-A. Storey, and K. Schneider. Mutual assessment in the social programmer ecosystem: an empirical investigation of developer profile aggregators. In *Proc. of the 2013 Conf. on Computer supported cooperative work, CSCW '13*, pages 103–116, 2013.
17. M. Squire. Should we move to Stack Overflow?: measuring the utility of social media for developer support. In *37th Intl. Conf. on Software Engineering*, pages 219–228, 2015.
18. I. Srba and M. Bielikova. Why is stack overflow failing? preserving sustainability in community question answering. *IEEE Software*, 33(4):80–89, 2016.
19. S. E. Stemler. A comparison of consensus, consistency, and measurement approaches to estimating interrater reliability. *Practical Assessment, Research & Evaluation*, 9, 4, 2004.

20. M.-A. Storey, L. Singer, B. Cleary, F. Figueira Filho, and A. Zagalsky. The (r) evolution of social media in software engineering. In *Proc. of the on Future of Software Engineering*, FOSE 2014, pages 100–116, 2014.
21. Y. R. Tausczik, A. Kittur, and R. E. Kraut. Collaborative problem solving: A study of mathoverflow. In *Proc. of the 17th ACM Conf. on Computer Supported Cooperative Work and Social Computing*, CSCW '14, pages 355–367, 2014.
22. B. Vasilescu. *Social aspects of collaboration in online software communities*. PhD thesis, Eindhoven University of Technology, 2014.
23. B. Vasilescu, A. Serebrenik, P. T. Devanbu, and V. Filkov. How social Q&A sites are changing knowledge sharing in open source software communities. In *Proc. of the 17th ACM Conf. on Computer Supported Cooperative Work and Social Computing*, pages 342–354, 2014.
24. Y. Yao, H. Tong, T. Xie, L. Akoglu, F. Xu, and J. Lu. Want a good answer? ask a good question first! *ArXiv e-prints*, 2013.
25. A. X. Zhang, M. S. Ackerman, and D. R. Karger. Mailing lists: Why are they still here, what is wrong with them, and how can we fix them? In *Proc. of the 33rd SIGCHI Conf. on Human Factors in Computing Systems*, 2015.