# A Comparison of Communication Tone and Responding Across Users and Developers in Two R Mailing Lists

Marc J. Lanovaz[1] and Bram Adams[2]
[1] *École de psychoéducation, Université de Montréal, Canada*
[2] *MCIS, Polytechnique Montréal, Canada*

***Abstract*—The R programming language has an active community of both users and developers, which maintain mailing lists to communicate. Given their differences in training and stability, the effects of communication tone on responding may differ across these two groups. We thus compared the prevalence and characteristics of different tones in the R-help user and R-devel developer mailing lists over a ten-year period as well as their relation to replies. Our analyses indicate that developers displayed marginally more positive and negative tones than users. Moreover, developers seemed less influenced by tone when choosing to reply to messages. Overall, our results suggest that different tones may produce small differences in responding across users and developers.**

***Keywords*-Sentiment Mining, Mailing Lists, Mining Software Repositories**

## I. Introduction

One of the fastest growing programming languages is R [1], which has an active community of both users and developers. R is a programming language that is used for statistical analyses, particularly by data scientists, academics, and health researchers. Interestingly, researchers have found that the R developer community (i.e., responsible for the development of the R core package) is more stable than its user community [2]. Given their differences in training and stability, the effects of communication tone on responding may also differ across users and developers. Comparing how communication tone affects responding appears important as both groups may require different communication strategies to promote active participation.

In recent years, researchers have developed tools to identify sentiment and emotions in text-based communication, such as SentiStrength-SE, Senti4SD, and EmoTxt [3]. The main limitation of these tools is that they typically perform best on the data sets on which they were trained [3], [4], [5]. Nevertheless, automated sentiment detection tools open many opportunities for researchers to efficiently measure communication tone in large data sets [6].

Apart from an early exploratory study [7], no researcher has validated the use of any of the previously mentioned tools with mailing lists. We selected SentiStrength-SE because it allowed us to categorize messages as having a positive or negative tone. In contrast, researchers designed EmoTxt and Senti4SD to return a specific emotion (e.g., anger, surprise, joy) [8], [9], which was not the intent of the current study. We examined emotions in two R mailing lists: the first list, R-help, targets mostly users whereas the second list, R-devel, is geared towards developers. Our research questions were:

- What is the prevalence of posts with negative and positive tones in the mailing lists?
- Do negative and positive posts differ in length and thread depth from neutral posts?
- Do replies differ based on the tone expressed in the initial post?
- Do results differ across users and developers?

## II. Our Approach

### A. The Data

We downloaded all emails published from 2008 to 2017 on the R-help and R-devel mailing lists [10], then used R's tm.plugin.mail package to parse the data and create a matrix containing each email's UNIX timestamp, number of characters, sentiment score (see below), thread number and thread depth. When extracting the email content and counting the number of characters, we removed any line starting with ">", "$" and , "[", as these symbols typically preceded text from the previous message or code output. After removing emails with no content, the R-help data set contained 235,309 email messages divided into 78,970 threads, while the R-devel data set had 25,771 emails divided into 7,354 threads. Our data and scripts are available on the Open Science Framework [11].

### B. Detecting Communication Tone

We used SentiStrength-SE to extract the tone expressed by the content of each of the messages. SentiStrength-SE is a lexical sentiment mining approach for software engineering-related documents. It provides two values that range from -4 (most negative) to +4 (most positive). The first value represents the most negative tone expressed by a word or short expression in the post and the second value the most positive tone. To obtain one sentiment score per email, we used SentiStrength-SE's scale output, which simply adds the two values together. To facilitate the analyses and to remain conservative in our classification, we considered a post as neutral if the scale output ranged from -1 to +1, as positive if the score was 2 or more, and as negative if the score was -2 or less.

## III. Calibrating SentiStrength-SE

To calibrate the SentiStrength-SE tool for our data sets, we first conducted a qualitative review of 100 randomly-selected posts (50 negative and 50 positive) from R-help to identify potential sources of misclassifications. The three most common causes of misclassifications for negative posts were related to the family name of a very active user ("Graves"; 4

cases), to the word "Poisson" being confused with "poison" (4 cases), and to the word "loss" (e.g., "packet loss"; 2 cases). For positive words, the expression "goodness of fit" led to three misclassifications and the word "fine" to two misclassifications. Moreover, we found three misclassifications due to emotional words in quotes following signatures.

To address the previous issues, we made the two following calibration changes prior to conducting our quantitative analysis. First, we changed the weights of the following words to zero in SentiStrength-SE: "grave", "poison", "loss", "goodness", and "fine". Second, we deleted the signatures and associated quotes in the email content by removing all text that followed the symbols " -- ". Following calibration, only 9 misclassifications remained in our randomly-selected posts, which is consistent with prior research on SentiStrength-SE [6].

## IV. Our Findings

### A. What is the Communication Tone in R-help and R-devel?

We began our analyses by examining the proportion of positive, negative, and neutral posts across each data set. In R-help, 95% of posts presented neutral tones. Only 2.3% and 3.1% of posts expressed negative and positive tones, respectively. In R-devel, neutral posts represented 92% of messages, with negative and positive posts each representing 4% of messages.

Interestingly, we found less emotional tones than other researchers in JIRA issue comments, StackOverflow posts, and code review comments [3], [9], [12]. A hypothesis for this discrepancy is that our criterion for categorizing a message as positive or negative may have been more stringent than for prior studies. Another hypothesis is that issue reports and review comments may be more conducive to producing emotional tones.

### B. Do Negative and Positive Posts Differ?

Next, we examined whether the length and thread depth of posts differed based on tone. The median lengths of messages for R-help and R-devel, respectively, were 339 and 471 characters for neutral posts, 574 and 723 for negatives posts, and 633 and 709 for positive posts. Our Kruskal-Wallis tests indicated that the difference across tones was statistically significant for both lists ($p < 0.0001$). In other words, negative and positive posts tended to be longer than neutral posts.

For all subsequent analyses, we only kept the initial emails in a thread ("depth 1") as well as all direct replies to an initial email ("depth 2"). Table I displays the distribution of posts across tones and both thread depths. The chi-square test conducted to examine the difference in tones across the two thread depths was statistically significant for R-help ($p < 0.0001$) and R-devel ($p = 0.008$). The distribution of posts across thread depths shows that, proportionally, posts with negative tones were less likely to be observed in initial posts than those with positive tones. This pattern was less pronounced in R-devel than in R-help.

### C. Do Replies Differ Across Tones?

Table II shows the distribution of the number of replies following an initial post, grouped by the tones in the initial post. The chi-square test for R-help showed a statistically significant difference in the distribution of replies across tones ($p < 0.0001$). The same test was not significant for the R-devel data. Closer examination of the R-help distribution shows that the largest difference was due to the negative posts being the more likely to receive no replies when compared to those that had neutral or positive tones.

For our next analysis, we examined the tone expressed in the first reply to an initial email. Prior to analysis, we removed single-post threads and all messages that did not contain a UNIX timestamp (which prevented the ordering of posts). Table III shows the results of this analysis. The chi-square tests were statistically significant for both lists ($p < 0.0001$). Regardless of the tone expressed in the initial post, the replies were overwhelmingly neutral. When emotional tones were displayed in replies, they were most likely to match the tone from the initial post.

### D. Do Results Differ Across Users and Developers?

The results remained generally consistent across users and developers. The main differences were (a) developers showed marginally more positive and negative tones, (b) the messages were longer in the R-devel list (regardless of emotion) and (c) the frequency distribution of replies differed. In the R-help list, negative posts were more likely to receive no reply. In contrast, the chi-square test was not significant for the R-devel mailing list. This discrepancy suggests that developers may be less influenced by tone when choosing to reply to messages.

## V. Threats to Validity

Our study has some threats to validity that should be noted. First, the design of our study was not experimental, which prevents us from determining the exact mechanisms responsible for the observed differences. At this point, we can only speculate as to why we observed differences across tones. Moreover, our parsing procedures remained imperfect despite our effort at removing signatures and previously quoted text. For example, signatures not preceded by " -- " were not removed. A threat to conclusion validity is that we used a single tool to detect emotions. Although we calibrated and validated SentiStrength-SE using a small subset of the posts, comparing the results of multiple sentiment detection tools would be relevant in future research. A final issue that merits further consideration is that we only conducted our study with two R mailing lists. Therefore, the extent to which our conclusions are applicable to other communities remains an open question.

## VI. Implications for Practice

As expected, most of the exchanges on the mailing lists were neutral, which is good news for developers in open source communities who rely substantially on mailing list communication for their daily operations. While a negative

Table I: Frequency distribution of thread depth by tones.

| | R-help | | | R-devel | | |
|---|---|---|---|---|---|---|
| | Tone | | | Tone | | |
| Thread Depth ↓ | Negative | Neutral | Positive | Negative | Neutral | Positive |
| 1 | 1,183 (23%) | 74,853 (35%) | 2,933 (42%) | 263 (29%) | 6,772 (31%) | 317 (35%) |
| 2 | 3,945 (77%) | 140,782 (65%) | 4,063 (58%) | 638 (71%) | 15,093 (69%) | 576 (65%) |

Note. The percentages are calculated by column.

Table II: Frequency distribution of the number of replies, grouped by tones in the initial post.

| | R-help | | | R-devel | | |
|---|---|---|---|---|---|---|
| | Tone | | | Tone | | |
| Number of Replies ↓ | Negative | Neutral | Positive | Negative | Neutral | Positive |
| 0 | 441 (37%) | 22,834 (30%) | 933 (32%) | 84 (32%) | 2,112 (31%) | 119 (37%) |
| 1 | 265 (23%) | 20,254 (27%) | 724 (25%) | 77 (29%) | 1,778 (26%) | 75 (24%) |
| 2 | 178 (15%) | 12,363 (17%) | 454 (15%) | 36 (14%) | 1,000 (15%) | 42 (13%) |
| 3 or more | 299 (25%) | 19,402 (26%) | 822 (28%) | 66 (25%) | 1,882 (28%) | 81 (26%) |

Note. The percentages are calculated by column.

Table III: Frequency distribution of tones in the first reply, grouped by tones in the initial post.

| | R-help | | | R-devel | | |
|---|---|---|---|---|---|---|
| | Tone of the Initial Post | | | Tone of the Initial Post | | |
| Tone of the First Reply ↓ | Negative | Neutral | Positive | Negative | Neutral | Positive |
| Negative | 56 (8%) | 965 (2%) | 45 (2%) | 16 (9%) | 144 (3%) | 4 (2%) |
| Neutral | 657 (90%) | 49,384 (96%) | 1,801 (91%) | 153 (87%) | 4,372 (95%) | 160 (82%) |
| Positive | 17 (2%) | 950 (2%) | 129 (7%) | 7 (4%) | 101 (2%) | 32 (16%) |

Note. The percentages are calculated by column.

tone decreased the likelihood of receiving a reply for users only, the tone of an email typically mimicked that of the initial message the email was replying to for both users and developers. As such, it makes sense for users and developers in open source communities to encourage neutral/positive tones in emails, either through guidelines or active moderation.

This paper also showed that a calibration may be important to obtain meaningful results. A recommendation for developers is to use a sample of messages to identify words or expressions that lead to misclassifications to calibrate sentiment detection tools prior to using them with novel data sets. Our results may thus help developers improve sentiment detection tools, or at least to specialize them to different communication media.

In sum, the comparison between R-help and R-devel showed how the presence of negative/positive tones can correlate with the type of audience and exchanged messages of a mailing list. Users and developers may use the results to improve the quality and flow of their asynchronous communications, which may potentially lead to increased participation in online communities. That said, the replication of our study with a wider range of projects and with different types of online communities [13] is key to examining the generality of our findings.

## References

[1] D. Robinson, "The impressive growth of r," https://stackoverflow.blog/2017/10/10/impressive-growth-r/, 2017.

[2] D. M. German, B. Adams, and A. E. Hassan, "The evolution of the r software ecosystem," in *2013 17th European Conference on Software Maintenance and Reengineering*. IEEE, 2013, pp. 243–252.

[3] M. R. Islam and M. F. Zibran, "A comparison of software engineering domain specific sentiment analysis tools," in *2018 IEEE 25th International Conference on Software Analysis, Evolution and Reengineering (SANER)*. IEEE, 2018, pp. 487–491.

[4] B. Lin, F. Zampetti, G. Bavota, M. Di Penta, M. Lanza, and R. Oliveto, "Sentiment analysis for software engineering: How far can we go?" in *2018 IEEE/ACM 40th International Conference on Software Engineering (ICSE)*. IEEE, 2018, pp. 94–104.

[5] R. Jongeling, P. Sarkar, S. Datta, and A. Serebrenik, "On negative results when using sentiment analysis tools for software engineering research," *Empirical Software Engineering*, vol. 22, no. 5, pp. 2543–2584, 2017.

[6] M. R. Islam and M. F. Zibran, "Leveraging automated sentiment analysis in software engineering," in *Mining Software Repositories (MSR), 2017 IEEE/ACM 14th International Conference on*. IEEE, 2017, pp. 203–214.

[7] P. Tourani, Y. Jiang, and B. Adams, "Monitoring sentiment in open source mailing lists - exploratory study on the apache ecosystem," in *Proceedings of the 2014 Conference of the Center for Advanced Studies on Collaborative Research (CASCON)*, Toronto, ON, Canada, November 2014, pp. 34–44.

[8] F. Calefato, F. Lanubile, and N. Novielli, "Emotxt: a toolkit for emotion recognition from text," in *2017 Seventh International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*. IEEE, 2017, pp. 79–80.

[9] F. Calefato, F. Lanubile, F. Maiorano, and N. Novielli, "Sentiment polarity detection for software development," *Empirical Software Engineering*, vol. 23, no. 3, pp. 1352–1382, 2018.

[10] "R: Mailing lists," https://www.r-project.org/mail.html, 2019.

[11] M. J. Lanovaz, "Data and r code," https://osf.io/ts5nq/?view_only=75387361aa184c18a794df5838346363, 2019.

[12] A. Murgia, P. Tourani, B. Adams, and M. Ortu, "Do developers feel emotions? an exploratory analysis of emotions in software artifacts," in *Proceedings of the 11th working conference on mining software repositories*. ACM, 2014, pp. 262–271.

[13] B. Lin, F. Zampetti, R. Oliveto, M. Di Penta, M. Lanza, and G. Bavota, "Two datasets for sentiment analysis in software engineering," in *2018 IEEE International Conference on Software Maintenance and Evolution (ICSME)*. IEEE, 2018, pp. 712–712.

## VII. Author Biographies

**Marc J. Lanovaz** is associate professor in the École de psychoéducation at the Université de Montréal. His research interests include the development and validation of software

technology to facilitate the delivery of services to individuals with developmental disabilities and their families. Lanovaz received his PhD in educational psychology from McGill University (Canada). Contact him at marc.lanovaz@umontreal.ca.

**Bram Adams** is associate professor at Polytechnique Montréal. His research interests include software release engineering, mining software repositories and the impact of human affect on software development. Adams received his PhD in computer science engineering from Ghent University (Belgium). Contact him at bram.adams@polymtl.ca.