Assessing Perceived Sentiment in Pull Requests with Emoji: Evidence from Tools and Developer Eye Movements

Kang-il Park
Department of Computer Science and Engineering
The University of Nebraska-Lincoln
Lincoln, Nebraska USA 68588
kangil.park@huskers.unl.edu

Bonita Sharif
Department of Computer Science and Engineering
The University of Nebraska-Lincoln
Lincoln, Nebraska USA 68588
bsharif@unl.edu

***Abstract*—The paper presents an eye tracking pilot study on understanding how developers read and assess sentiment in twenty-four GitHub pull requests containing emoji randomly selected from five different open source applications. Gaze data was collected on various elements of the pull request page in Google Chrome while the developers were tasked with determining perceived sentiment. The developer perceived sentiment was compared with sentiment output from five state-of-the-art sentiment analysis tools. SentiStrength-SE had the highest performance, with 55.56% of its predictions being agreed upon by study participants. On the other hand, Stanford CoreNLP fared the worst, with only 5.56% of its predictions matching that of the participants'. Gaze data shows the top three areas that developers looked at the most were the comment body, added lines of code, and username (the person writing the comment). The results also show high attention given to emoji in the pull request comment body compared to the rest of the comment text. These results can help provide additional guidelines on the pull request review process.**

***Index Terms*—sentiment analysis, pull requests, empirical study, sentiment tools, eye tracking study, emoji, comments**

I. Introduction

Software developers express sentiment through many different means and artifacts. Source code and commits have been studied for sentiment in the past [1] and were shown to have largely neutral sentiment and negative results. This is not surprising since there is no direct collaboration between developers when writing the code, causing most things to be reported as neutral. The collaboration happens elsewhere in issue trackers, mailing lists, emails, and other mediums. In this paper, we seek to study one particular medium where developers might express sentiment i.e., the pull request (PR) on GitHub. The pull request provides a means for developers to talk to each other about the code involved. This is in contrast to finding sentiment in commit messages. Since emoji are used quite often in online discourse [2] and increasingly within GitHub [3], they might also play a role in how developers perceive sentiment.

One way to determine how much attention a developer gives to parts of a PR page is to observe what they are looking at on the page while they are doing the task. In order to do this, we conduct a pilot eye tracking study where we continuously monitor developer eye gaze while they read and evaluate the PR for any sentiment. This gives us an objective view of what the developer read while on task.

To study the attention developers give to PR items while assessing perceived sentiment, we present a pilot eye tracking study conducted in Google Chrome while seamlessly collecting data using community infrastructure provided by iTrace [4]. This study seeks to answer two research questions:

- RQ1 How do current state-of-the-art sentiment analysis tools compare against perceived developer sentiment in GitHub PRs with emoji?
- RQ2 What elements of the PR page do developers spend most of their attention on when reading a PRs with emoji in the context of analyzing sentiment?

The first research question seeks to understand how multiple state-of-the-art sentiment analysis tools compare against perceived developer sentiment on a set of twenty-four GitHub PRs. We were interested in determining if PRs fare differently from what was observed in sentiment on commits in prior work [1]. The second research question leverages the fine-grained eye movement data collected on the PR page within Google Chrome to determine exactly which elements developers read when tasked with analyzing sentiment. Note that even though developers are not tasked with analyzing sentiment in PRs as a typical software engineering (SE) task, it does act as a precursor to how developers feel within the team environment and is hence relevant to study. We chose these RQs because we were interested in learning how tools rate PRs (RQ1) compared to how developers focus their gaze attention to rate PRs for sentiment (RQ2).

The main contribution of this paper is a pilot eye tracking study conducted on GitHub PR pages where the developer is able to scroll and read through the comments. It has been shown in the past that studies in realistic environments produce different results than in unrealistic single static screen scenarios [5]. As far as we know, this is the first realistic eye tracking conducted on GitHub PR pages in the context of assessing

perceived sentiment. The main results show SentiStrength-SE agreeing 55% of the time with perceived developer ratings of sentiment. In addition, emojis were given greater attention than other comment text, indicating that perhaps this practice should be encouraged and used more consistently. We believe the results of this study could help towards the overarching goal of providing additional guidelines on PR reviewing.

II. Related Work

Novielli et al. [6] emphasized the domain-dependent nature of sentiment analysis by running 100 Stack Overflow questions through SentiStrength. SentiStrength [7] (a general domain tool) detects positive and negative sentiment with a strength scale of 1-5. However, domain-specific communication is harder to classify with a general tool. Previous studies only focused on identifying product opinions rather than user behaviors in social media with short, informal text. Islam et al. [8] conducted a qualitative study to identify the reasons why sentiment analysis tools such as SentiStrength and NLTK perform worse in a software engineering context. The authors determined that domain-specific variations were the biggest challenge. To address this, the authors proposed a modified version of SentiStrength - SentiStrength-SE, a sentiment analysis tool that uses domain-specific dictionary and heuristics based on a "Gold Standard" dataset of 5,600 issue comments from the JIRA issue tracking system. By narrowing the scope to a specific domain, SentiStrength-SE was able to have an overall average accuracy of up to 81% versus 66% for SentiStrength.

One of the studies using SentiStrength-SE by Huq et al. [9] conducted a statistical study on GitHub PRs, as opposed to JIRA. The study aimed to see the difference in sentiment based on whether or not a PR introduced a Fix-Inducing Change (FIC), or code that introduces bugs to a given system, inducing its fix in the future. Six GitHub repositories' PRs were extracted and run through SentiStrength-SE to rate a sentiment of either commits, comments, reviews, or all components of a given PR. The study concluded that the general sentiment in PRs with FICs is more negative (28.5% neg. vs. 59.5% pos.) than those without FICs (17.9% neg. vs. 33.2% pos.).

Ahmed et al. [10] propose SentiCR that uses the Gradient Boosting Tree (GBT) method. A new dataset was generated to compare its accuracy against other sentiment analysis tools from an eight-step approach to mine 1600 code review comments from repositories of 20 popular open-source projects in Gerrit. While SentiCR can predict sentiment with up to 83% accuracy versus 69% and 72% of SentiStrength and NLTK, respectively, it only displays results in a single polarity rather than both positive and negative.

Lu et al. [3] studied the usage and sentiment of emoji by developers on GitHub. Their study also aimed to determine how other developers interpret these already written issues, PRs, and comments using text representation learning methods to determine an emoji's possible meaning. The intention of emoji usage is also analyzed to ultimately obtain a greater understanding of software engineering culture by categorizing use based on a set of intention categories. However, the actual impact of emoji usage on other developers and the sentiment of further replies were not investigated. They also did not look into how these were read by developers. The study presented in this paper is the first eye tracking study on how developers read and perceive sentiment in GitHub PRs. The perceived sentiment is then compared to tool output.

III. Pilot Study Design

This study compares the interpreted sentiment of human participants on GitHub PRs against a variety of sentiment analysis tools while using i-Trace Core [4] and its Chrome plugin for recording eye movements. The PRs all have emoji in their discussions because we wanted to determine the significance of their usage in perceived emotion and attention given to them during reading. Note that current sentiment analysis tools do not take emoji into consideration when predicting sentiment. The replication package is available at https://zenodo.org/record/4602631.

A. Participants

Six volunteers (four females and two males) were recruited from a local university both verbally and via email. There were two undergraduate students, two graduate students, and two professionals from industry. Three participants (50%) responded in the post-questionnaire that they were familiar with reviewing GitHub PRs. Out of the three, one reviewed PRs often, while the other 2 occasionally.

B. Eye Tracking Apparatus and Environment

While each participant was performing their task, their eye movements were recorded by the Tobii Pro TX300 eye tracker running at 60Hz. The eye tracker was run in tandem with a combination of i-Trace Core (server) [4] and a version of its Chrome plugin (see Figure 1) that is customized to track any gazes on emojis, comment body text, usernames, and other items on the PR page.

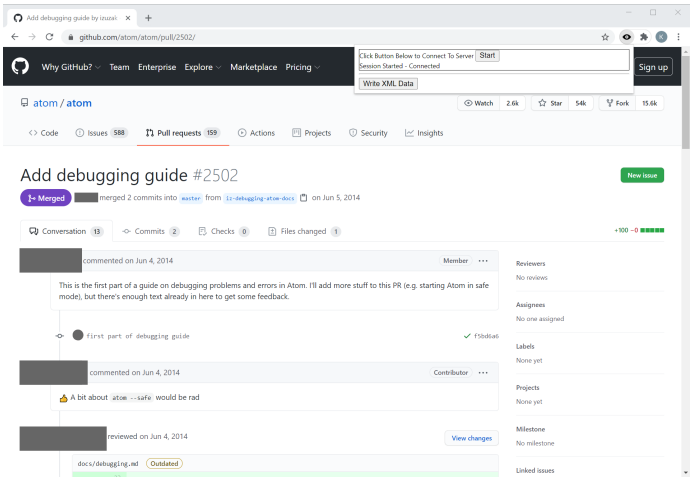

Fig. 1. i-Trace Chrome's GUI while browsing a GitHub PR page showing the start tracking button on top and the writing of XML gaze data functionality.

C. PR Selection - Subject Systems Used

We selected 44 projects randomly that returned any use of emojis and had over 100 stars to avoid trivial projects. The projects were further narrowed down by reading its PR pages to determine whether or not emojis were frequently used in conversation. A subset of the publicly available GHTorrent dataset [11] was generated. From this subset, 24 PRs were manually selected to ensure a given PR was a genuine conversation and included at least one emoji in a conversational comment, for an average of 1.08 emoji per PR. The relatively small sample size of PRs was chosen to allow developers to read and rate all comments within a reasonable time. The PRs are from a variety of open-source projects on GitHub with a variable number of comments per PR, ranging from 2 to 38 comments. The PRs also vary in the programming language used: JavaScript, CoffeeScript, Ruby, etc., but the main focus is on the sentiment of the comments written themselves that are written in plain English.

D. Sentiment Analysis Tools Used

The sentiment analysis tools analyzed in this study are: SentiStrength [7], SentiStrength-SE [8], SentiCR [10], NLTK [12], and Stanford-NLP [13]. Of these tools, SentiStrength-SE and SentiCR are designed specifically for sentiment analysis in the software engineering domain. We ran the tools on all 24 GitHub PR comment conversations.

E. Study Variables

The eye tracking dependent variables are defined in the context of an area of interest (AOI). The I-VT fixation filter [14] is run on all raw gazes to determine gaze fixations (i.e., stabilization of the eyes for a period of time - threshold was set to 60 ms). The dependent variables are given below.

- Developer perceived sentiment via an online questionnaire (positive, negative, neutral)
- Percentage of sentiment analysis predictions agreeing with developer perceived sentiment
- Adjusted fixation duration (AFD) - fixation duration normalized relative to how long an AOI is
- First fixation duration (FFD) - duration of the first fixation on an AOI
- Single fixation duration (SFD) - duration of a single fixation on an AOI if exists
- Total duration (TD) - sum of fixation durations on an AOI
- Fixation count (FC) - number of fixations on an AOI
- Duration of emoji fixations vs. other AOIs
- Adjusted duration of emoji fixations vs. comment text

The FFD, SFD, TD, and FC metrics were inspired from [15]. See Section IV for adjusted durations.

F. Study Instrumentation

Prior to the study, developers completed a pre-study questionnaire on their software engineering proficiency. They were then calibrated for the eye tracker and shown one PR at a time, after which they completed their perceived assessment of positive, negative, or neutral. This process was repeated for all 24 PRs. The study was conducted in a lab setting on an extended monitor display with a moderator present to set up the study and tasks in the Chrome browser. For a given task, participants were asked to read a PR and record their responses. The expected time to completion for the entire study by a participant ranged from 45 minutes to an hour total. They were given as much time as they needed to complete the tasks. Finally, participants answered a post-study questionnaire that gathered information to improve future studies as well as gauged the level of familiarity with the use of the GitHub platform. We intentionally asked this question in the post-survey to avoid any performance biases.

IV. Post-processing and Normalization

Each of the 24 PRs used for the study was downloaded as HTML files and processed through a post-processing script in order to analyze a participant's eye movements by areas of interest (AOIs), such as the individual comments within a PR, versus analyzing the PRs holistically as was the case with sentiment analysis tools. This script's output is a CSV file recording the length of all PR comments to derive the adjusted duration.

In order to determine what developers were looking at and detect fixations (stabilization of the eyes for a certain amount of time), we run the I-VT fixation filter (velocity-based filter) on all the raw gazes generated in the XML files generated after the completion of each task. This filter generates a CSV file with all the filtered fixations as well as their duration in ms, which is then combined with the results from the HTML post-processing script to derive the adjusted duration as:

$$adj.duration = \frac{duration * total\ characters}{character\ count\ of\ AOI\ type}$$

where the adjusted duration was derived in order to normalize the gaze duration relative to how long a given comment is.

Despite the fact that an emoji is considered to be one encoded character in Unicode, humans do not perceive emojis with the same amount of weight as a machine would. Cohn et al. [16] conducted a study with 72 participants comparing the reading times between a sentence with pure text and the same sentence with an emoji replacing either a noun or a verb in that sentence. Replacing the words in a sentence with emoji was determined to make reading times longer while not affecting readability at the same time. A study from Robus et al. [15] used eye-tracking equipment while determining the effects of perceived sentiment by users. In this study, both emojis and five-letter target words were AOIs that resulted in similar gaze times, regardless of the emoji's sentiment. Therefore, we normalize emojis as being equivalent to a word with 5 characters in this study.

V. Experimental Analysis and Results

A. RQ1 Results: Sentiment Analysis Tool Predictions vs. Developer Perceived Sentiment on PRs

Overall, participant responses to perceived sentiment in PRs show that humans only rated 5.56% of all PRs as negative,

with the total responses being 68 positive, 8 negative, and 68 neutral. In comparison, the tools' percentage of negative predictions are 19.2%(SentiStrength), 9.29%(SentiStrength-SE), 14.43%(SentiCR), 50.79%(NLTK), and 71.29%(Stanford CoreNLP). Of note are SentiStrength-SE and SentiCR, the two tools selected for this study designed for the software engineering domain, which are the two lowest percentages for negative predictions.

The current state of the art sentiment analysis tools has their overall predictions on a given PR displayed against human participant responses in Table I, which are compared against human results as a percentage of how many human responses agree with a given tool for each PR. Of note in Table I are PRs #1776 and #2291, where every single tool predicted the PR to be negative. However, participants primarily interpreted the PRs as neutral, 3 for #2291 and 5 for #1776, resulting in 0% for every column in that row. As shown in Figure 2, the conversation can be interpreted as a positive one, where two developers are giving each other friendly remarks, such as "Good call" or "Ah you're totally right. If only I could type 😄." The results overall were not favorable to the tools used in the study: SentiStrength, SentiStrength-SE, SentiCR, NLTK, and Stanford CoreNLP gave 50.69%, 55.56%, 20.14%, 25.69%, and 5.56% accuracy against human participants' results, respectively. While the accuracy has taken a great hit due to the lack of consideration for emojis in these tools, Stanford CoreNLP has the lowest accuracy overall in predicting sentiment due to the training data not being in the software engineering domain.

TABLE I
PERCENTAGE OF PULL REQUEST PREDICTIONS AGREEING WITH PARTICIPANT RESPONSES PER PULL REQUEST

PR#	SentiStr	SentiStr-SE	SentiCR	NLTK	Stanford NLP
97	67% Neu	33% Pos	0% Neg	67% Neu	0% Neg
209	83% Pos	83% Pos	0% Neg	17% Neu	0% Neg
299	100% Pos	0% Neu	0% Neg	0% Neg	0% Neg
346	17% Pos	17% Neg	17% Neg	17% Neg	17% Neg
364	50% Pos	50% Pos	0% Neg	0% Neg	0% Neg
408	33% Pos	67% Neu	0% Neg	33% Pos	0% Neg
415	83% Pos	83% Pos	17% Neu	17% Neu	0% Neg
441	33% Pos	33% Pos	0% Neg	67% Neu	0% Neg
685	0% Neg	100% Neu	100% Neu	0% Neg	0% Neg
1683	33% Pos	67% Neu	0% Neg	0% Neg	0% Neg
1776	33% Neg	33% Neg	33% Neg	33% Neg	33% Neg
1794	67% Neu	67% Neu	67% Neu	67% Neu	0% Neg
2291	0% Neg	0% Neg	0% Neg	0% Neg	0% Neg
2426	17% Pos	67% Neu	17% Neg	17% Neg	17% Neg
2457	50% Neu	50% Neu	17% Neg	17% Neg	17% Neg
2502	100% Pos	0% Neu	0% Neg	100% Pos	0% Neg
2603	83% Pos	83% Pos	17% Neu	0% Neg	0% Neg
3142	33% Pos	67% Neu	0% Neg	0% Neg	0% Neg
3241	0% Neg	100% Neu	100% Neu	0% Neg	0% Neg
3394	50% Pos	50% Pos	50% Neu	0% Neg	0% Neg
3433	83% Pos	83% Pos	0% Neg	0% Neg	0% Neg
3633	83% Pos	83% Pos	0% Neg	17% Neu	0% Neg
3697	17% Pos	17% Pos	50% Neg	50% Neg	50% Neg
4973	100% Pos	100% Pos	0% Neu	100% Pos	0% Neu
Avg	50.69%	55.56%	20.14%	25.69%	5.56%
Std. Dev	7.65%	5.05%	6.13%	10.01%	6.27%

Fig. 2. Excerpt of PR #2291 from Code.org's repository used in the study.

To further look into the low accuracy of sentiment analysis tools, individual participant results versus tool results were examined. While the averages returned similar numbers when results are separated by participant, there are noticeably more rows where all tools did not match with what a given participant responded with, as only one PR, #2291, returned 0% for every tool in the aggregated results. For example, participant P1 responded Neutral, Neutral, Positive, Neutral, and Neutral for PR numbers 346, 1776, 1794, 2291, and 3697, respectively. All of the participant's results did not match what was predicted by the sentiment analysis tools for those given PRs. The average number of PRs that disagree with a participant for all tools is 4.

B. RQ 2 Results: Gaze Attention Breakdown on PRs

Based on the 5 participants that had their eye movements recorded while looking at a given PR, the frequency of fixations per AOI by the participants has been recorded in Table II. One participant's eye tracking data could not be used due to data loss. As the participants were asked to read the conversations in PR comment pages, the comment body had 4317 total fixations. When comparing fixation duration per AOI as shown in Figure 3, one AOI that was noticeable was the use of emoji in PR comments. When they are compared to other AOIs, the non-adjusted fixations on emoji are on average longer by 98.9%. Another thing to note is despite emojis' relative sparseness in PR comments versus the rest of the AOIs on the page as shown in Figure 2, they are close to the table's median at 33 total fixations. However, a simple comparison of duration does not take into account the fact that different AOIs have differing lengths, and adjusting the results by normalizing them relative to the length of the AOI accounts for this. One emoji is considered equivalent to a 5 character word for this study, as Cohn et al.'s study [16] concluded that, after giving emojis equivalent weight to a 5 character word, fixation duration of emoji did not change regardless of what emotion the sentence or the emoji itself expressed. The average adjusted duration show that, relative to its length, gazes on emoji are significantly longer (18.4x) than a gaze on a comment body, as shown in Figures 4 and 5,

where Figure 4 is the average adjusted duration for the entire study whereas Figure 5 shows the average adjusted duration for each participant. To further determine how the adjusted duration of an emoji is an order of magnitude higher than the body(42337.13 vs. 2300.01) but the number of gazes on the emoji themselves is less than 1/100th(4317 vs. 33) of the number of gazes on the comment body, further metrics were calculated as seen in Table III, where the average total duration of a comment is longer than that of an emoji's(168984.02 vs. 73739.12), but the average first fixation duration of an emoji is significantly longer (42337.13 vs. 552.48). On the other hand, every comment had more than one fixation, and therefore, there were no single fixations on a comment body AOI.

TABLE II
FREQUENCY OF FIXATIONS PER AOI

AOIs	Freq
Labels	1
Num of Commits	1
Projects	1
Reviewers	1
Assignees	2
Num of Conversation Comments	2
Participation	4
Issue Label	5
Emoji	33
File	54
Deleted LOC	93
Commit Name	134
Unchanged LOC	146
PR Title	332
Comment Date	349
Image	524
Username	546
Added LOC	661
Comment Text	4317

TABLE III
OBSERVATIONS, MEANS, AND STANDARD DEVIATIONS OF MEASURES BY CONDITIONS.

AOI	Obs	Mean	Std Dev
Emoji			
FFD	11	48609.2	46717.17
SFD	3	65917	33214.02
TD	18	77618.067	84460.62
FC	33	42337.13	43821.77
Comment Text			
FFD	15	552.48	839.81
SFD	0	NA	NA
TD	61	162773.09	324792.72
FC	4317	2300.01	6900.21

VI. DISCUSSION

With respect to RQ1, because the current state of the art sentiment analysis tools does not take into consideration the usage of emoji, their predicted sentiment on GitHub PRs with emoji does not accurately align with interpretations by human participants, going as low as 5.83% for the sentiment predicted vs. human responses in the case of Stanford CoreNLP. Despite the fact that conversational messages can have their expressed

Overview	P1	P2	P3	P4	P5	Avg
Emoji	129.50	232.33	289.60	230.14	257.50	227.82
File	71.50	242.42	255.12	166.25	136.07	174.27
Image	130.07	142.58	192.60	170.13	161.91	159.46
Username	117.08	159.15	157.20	231.21	121.30	157.19
Deleted LOC	112.03	149.33	384.87	0.00	136.96	156.64
Unchanged LOC	120.55	149.34	158.48	207.50	131.75	153.52
Commit Name	103.15	213.04	173.42	143.00	123.51	151.22
Comment Text	113.57	170.21	184.57	149.38	133.72	150.29
PR Title	117.83	179.52	237.91	133.00	83.00	150.25
Other	112.02	157.29	183.93	168.65	128.38	150.05
Added LOC	112.99	172.43	171.40	123.21	146.28	145.26
Comment Date	115.67	157.74	156.10	163.57	113.15	141.25
Num of Conversation Comments	0.00	0.00	83.00	449.00	0.00	106.40
Participation	0.00	149.50	149.00	0.00	83.00	76.30
Issue Label	0.00	0.00	149.00	0.00	82.67	46.33
Labels	0.00	0.00	183.00	0.00	0.00	36.60
Num of Commits	0.00	149.00	0.00	0.00	0.00	29.80
Projects	0.00	149.00	0.00	0.00	0.00	29.80
Assignees	0.00	141.00	0.00	0.00	0.00	28.20
Reviewers	0.00	99.00	0.00	0.00	0.00	19.80

Fig. 3. Average fixation non-adjusted duration in ms for every AOI per participant

Type	Avg Adj. Duration
Comment Text	2300.01
Emoji	42337.13

Fig. 4. Average adjusted duration (ms) for comment and emoji AOIs

sentiment be more explicit through the use of emoji, sentiment analysis tools not being able to interpret this key part of the text prevents it from providing accurate results, with significantly worsening results in tools that are not designed for the software engineering domain in particular.

With respect to RQ2, as GitHub PR comment pages are web pages where developers discuss the contents of a given PR, the conversational comment bodies are the most looked at element in a given PR. However, due to emoji's pictographic nature, developers looking through a PR comment are more

Overview	P1	P2	P3	P4	P5	Avg
Emoji	53013.28	55101.27	20427.68	19701.49	71780.10	44004.76
Comment Text	1853.23	2923.53	3065.33	904.04	1479.71	2045.17

Fig. 5. Average adjusted duration(ms) for comment and emoji AOIs per participant

likely to spend more time looking at the emoji vs. other words that are within that message disproportionately. Even with the equivalence in length determined by [16] that weighs emojis as five characters regardless of the sentiment expressed within the emoji itself, developers that participated in this study looked at emojis far longer than any other word within a PR comment. These results could be used by companies to provide additional guidelines on how to write PRs suggesting the use of emoji when needed.

VII. Threats to Validity

There may not be enough repositories used for the study to be representative of all of GitHub. However, in order to mitigate this threat, all projects are active and have at least 100 stars so that smaller repositories such as abandoned personal projects are not selected. In order to provide consistency and ease of replication for the study, all sentiment analysis tools are run with out-of-the-box default settings, despite multiple settings and parameters being available for each. However, settings customized for the data of this study can negatively affect the accuracy of developer comments not included in the study.

VIII. Conclusions and Future Work

The study compares multiple state-of-the-art sentiment analysis tools against developers' perceived sentiment in interpreting GitHub PRs containing emojis. While current tools consider the usage of emoticons when predicting expressed sentiment, the newer and increasingly more prevalent use of emoji is not considered. Despite the promising results shown in prior work, this introduction of emoji and the tools' lack of consideration has proven to widen the discrepancy between tool predictions and human interpretations.

The results of this pilot study show that the percentage of predictions agreeing with human participants ranges from 55.56% with SentiStrength-SE to a meager 5.56% of Stanford NLP. The increase in discrepancies can be explained with eye tracking information that shows when a developer looks at an emoji; the fixation is disproportionately higher than the rest of the text in GitHub PR comments where emojis fixations are roughly 18.4 times that of other comment body text after normalization for length.

The insights provided by this study offer potential future work that proposes improved sentiment analysis tools that take both into consideration communication in the domain of software engineering and the use of emoji. This study can be expanded upon by increasing the sample size of developers and expanding the list of tools, such as Senti4SD [17] or SentiMoji [18], a recent tool that uses emoji and trained in the software engineering domain. In such a tool, comparisons can also be made by observing the changes in interpreted sentiment based on the inclusion or omission of emoji in the same comment. Future studies may also take a more in-depth insight into how sentiment in a communication sentence between developers may be affected in proportion to how long a reader looks at the emojis present.

References

[1] R. Jongeling, P. Sarkar, S. Datta, and A. Serebrenik, "On negative results when using sentiment analysis tools for software engineering research," *Empir. Softw. Eng.*, vol. 22, no. 5, pp. 2543–2584, 2017. [Online]. Available: https://doi.org/10.1007/s10664-016-9493-x

[2] W. Brants, B. Sharif, and A. Serebrenik, "Assessing the meaning of emojis for emotional awareness-a pilot study," in *Companion Proceedings of The 2019 World Wide Web Conference*, 2019, pp. 419–423.

[3] X. Lu, Y. Cao, Z. Chen, and X. Liu, "A first look at emoji usage on github: An empirical study," *arXiv preprint arXiv:1812.04863*, 2018.

[4] D. T. Guarnera, C. A. Bryant, A. Mishra, J. I. Maletic, and B. Sharif, "itrace: Eye tracking infrastructure for development environments," in *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications*, ser. ETRA '18. New York, NY, USA: ACM, 2018, pp. 105:1–105:3. [Online]. Available: http://doi.acm.org/10.1145/3204493.3208343

[5] N. J. Abid, B. Sharif, N. Dragan, H. Alrasheed, and J. I. Maletic, "Developer reading behavior while summarizing java methods: size and context matters," in *Proceedings of the 41st International Conference on Software Engineering, ICSE 2019, Montreal, QC, Canada, May 25-31, 2019*, J. M. Atlee, T. Bultan, and J. Whittle, Eds. IEEE / ACM, 2019, pp. 384–395. [Online]. Available: https://doi.org/10.1109/ICSE.2019.00052

[6] N. Novielli, F. Calefato, and F. Lanubile, "The challenges of sentiment detection in the social programmer ecosystem," in *Proceedings of the 7th International Workshop on Social Software Engineering*, 2015, pp. 33–40.

[7] M. Thelwall, K. Buckley, G. Paltoglou, D. Cai, and A. Kappas, "Sentiment strength detection in short informal text," *Journal of the American Society for Information Science and Technology*, vol. 61, no. 12, pp. 2544–2558, 2010.

[8] M. R. Islam and M. F. Zibran, "Sentistrength-se: Exploiting domain specificity for improved sentiment analysis in software engineering text," *Journal of Systems and Software*, vol. 145, pp. 125–146, 2018.

[9] S. F. Huq, A. Z. Sadiq, and K. Sakib, "Understanding the effect of developer sentiment on fix-inducing changes: An exploratory study on github pull requests," in *2019 26th Asia-Pacific Software Engineering Conference (APSEC)*. IEEE, 2019, pp. 514–521.

[10] T. Ahmed, A. Bosu, A. Iqbal, and S. Rahimi, "Senticr: a customized sentiment analysis tool for code review interactions," in *Proceedings of the 32nd ieee/acm international conference on automated software engineering*. IEEE Press, 2017, pp. 106–111.

[11] G. Gousios, "The ghtorent dataset and tool suite," in *Proceedings of the 10th working conference on mining software repositories*. IEEE Press, 2013, pp. 233–236.

[12] S. Bird, E. Klein, and E. Loper, *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.", 2009.

[13] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the 2013 conference on empirical methods in natural language processing*, 2013, pp. 1631–1642.

[14] R. Andersson, L. Larsson, K. Holmqvist, M. Stridh, and M. Nyström, "One algorithm to rule them all? an evaluation and discussion of ten eye movement event-detection algorithms," *Behavior research methods*, vol. 49, no. 2, pp. 616–637, 2017.

[15] C. M. Robus, C. J. Hand, R. Filik, and M. Pitchford, "Investigating effects of emoji on neutral narrative text: Evidence from eye movements and perceived emotional valence," *Computers in Human Behavior*, p. 106361, 2020.

[16] N. Cohn, T. Roijackers, R. Schaap, and J. Engelen, "Are emoji a poor substitute for words? sentence processing with emoji substitutions." in *CogSci*, 2018.

[17] 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.

[18] Z. Chen, Y. Cao, H. Yao, X. Lu, X. Peng, H. Mei, and X. Liu, "Emoji-powered sentiment and emotion detection from software developers' communication data," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 30, no. 2, pp. 1–48, 2021.