A Comparison of Software Engineering Domain Specific Sentiment Analysis Tools

Md Rakibul Islam
University of New Orleans, USA
mislam3@uno.edu

Minhaz F. Zibran
University of New Orleans, USA
zibran@cs.uno.edu

***Abstract*—Sentiment Analysis (SA) in software engineering (SE) text has drawn immense interests recently. The poor performance of general-purpose SA tools, when operated on SE text, has led to recent emergence of domain-specific SA tools especially designed for SE text. However, these domain-specific tools were tested on single dataset and their performances were compared mainly against general-purpose tools. Thus, two things remain unclear: (i) how well these tools really work on other datasets, and (ii) which tool to choose in which context. To address these concerns, we operate three recent domain-specific SA tools on three separate datasets. Using standard accuracy measurement metrics, we compute and compare their accuracies in the detection of sentiments in SE text.**

I. Introduction

Sentiment Analysis (SA) in software engineering (SE) text has recently drawn interests in the community [12], [13]. Earlier attempts used general-purpose (i.e., domain independent) sentiment detection tools (e.g., `SentiStrength` [25], `NLTK` [1] and `Stanford NLP` [2]) for SA in SE text. Those general purpose SA tools are found to have very low accuracies when operated on text from a technical domain such as software engineering [12], [16], [26]. Those SA tools were developed and trained using data from non-technical social networking media (e.g., twitter posts, forum posts, movie reviews) and perform poorly for software engineering text largely due to domain specific variations in meanings of frequently used technical terms [15].

Thus, recent attempts have led to the development of a few domain specific SA tools especially designed to deal with SE text. Each of these domain specific SA tools were originally evaluated using a different dataset and compared against the existing domain independent SA tools. The datasets (e.g., JIRA issue comments, Stack Overflow posts, code review comments) differ in the proportion and category of technical text they include. The accuracies of these SE domain specific tools have never been compared using multiple datasets.

Using multiple datasets, we carry out a quantitative comparison of three recently released SE domain specific SA tools. In our study, we address the following two research questions. **RQ1**: *Can we identify a tool, which shows the highest accuracy across different datasets*? — We investigate which tool achieved higher accuracy in which dataset, and we distinguish a tool which achieves overall the best accuracy across all the datasets. This will help one in choosing the most appropriate tool for SA in SE text.

RQ2: *To what extent do the different sentiment analysis tools (dis)agree with each other*? —Here we examine to what extent the SA tools (dis)agree on their detection of sentimental polarities (i.e., positivity, negativity, and neutrality) in SE text. This agreement analysis will help in identifying the spots where those tools might need improvements.

II. Datasets

In our study, we use three ground-truth datasets drawn from software development ecosystems. A summary of these three datasets is presented in Table I.

TABLE I
Summary of the Datasets Used in this Study

Dataset	Group	# of Comments				
		Total	Pos	Neg	Neu	Non-neg
JIRA Issue Comments	Group-2	1,576	748	128	700	1,448
	Group-3	4,000	375	672	2,953	3,324
SOP	NA	4,423	1,527	1,202	1,694	3,221
CRC	NA	2,000	NA	398	NA	1,202

A. JIRA Issue Comments (JIC) Dataset

This dataset is based on the work of Ortu et al. [22]. The entire dataset is divided in three groups named as Group-1, Group-2, and Group-3. Group-1 contains 392 issue comments and Group-2 contains 1,600 issue comments. Group-3 contains 4,000 sentences written by developers. Each individual text (i.e., issue comments and sentences) in the dataset are manually annotated with emotions such as *love, joy, surprise, anger, sadness* and *fear*. For manual annotation, each of the 5,992 individual text is interpreted by n distinct human raters [22] and annotated with emotional expressions as found in those comments. For Group-1, $n = 4$ while for Group-2 and Group-3, $n = 3$. In this study, we use the Group-2 and Group-3 portions of the dataset as those were also used in other studies [15], [22].

1) Emotional Expressions to Sentimental Polarities: We compute sentimental polarities (i.e., positivity, negativity, and neutrality) from the emotional expressions (i.e., *love, joy, surprise, anger, sadness, fear*) as follows. Emotional expressions joy and $love$ denote *positive* sentiment, while $anger$, $sadness$, and $fear$ indicate *negative* sentiment. We take special measurement for the *surprise* expressions as in some cases, an expression of $surprise$ can indicate positive polarity, denoted as $surprise^{+}$, while in other cases it can express a *negative* sentiment, denoted as $surprise^{-}$. Thus the issue

978-1-5386-4969-5/18/$31.00 © 2018 IEEE

SANER 2018, Campobasso, Italy
ERA Track

comments in the benchmark dataset, which are annotated with *surprise* emotion, need to be further classified based on the sentimental polarities they convey. Hence, we get each of such comments interpreted by three human raters (computer science graduate students), who independently assign polarities of the surprise expressions in each comments.

We consider a *surprise* expression in a comment negatively polarized (or positively), if two of the three raters identify negative (or positive) polarity in it. We found 79 issue comments in the benchmark dataset, which were annotated with the *surprise* expression. 23 of them express *surprise* with positive polarity and the rest 56 convey negative *surprise*.

Then we split the set $\mathcal{E}$ of emotional expressions into two disjoint sets as $\mathcal{E}_+ = \{joy, love, surprise^+\}$ and $\mathcal{E}_- = \{anger, sad, fear, surprise^-\}$. Thus, $\mathcal{E}_+$ contains only the positive sentimental expressions and $\mathcal{E}_-$ contains only the negative sentimental expressions. A similar approach is also used in other work [15], [16], [17] to categorize emotional expressions according to their polarities.

2) Assignment of Sentiments to Text: A piece of text is assigned positive sentiment if maximum number of raters among the n raters identify positive sentiment in that post. For example, in Group-2, a text $\mathcal{T}$ is considered to have a positive sentiment, if two of the three raters agree on perceiving positive sentiment in $\mathcal{T}$. Similarly, negativity and neutrality of pieces of text are also determined based on majority agreements. We find that human raters could not agree on the sentiments of 24 issue comments in Group-2. These 24 comments are excluded from this study.

B. Stack Overflow Posts (SOP) Dataset

The second ground-truth dataset we use is based on the work of Calefato et al. [6]. This dataset is composed of 4,423 posts from Stack Overflow. Each if the 4,423 posts is interpreted and annotated with sentimental polarities (i.e., positive, negative, neutral) by three distinct human raters and a total 12 different raters were used to annotate the entire dataset. The sentiments expressed in a particular post are determined based on majority agreements. Thus, in this dataset, 35% of posts convey positive sentiment and 27% express negative sentiment while 38% of posts are neutral in sentiments.

C. Code Review Comments (CRC) Dataset

The third dataset used in this work is based on the work of Ahmed et al. [4]. This dataset contains manually annotated 2,000 code review comments drawn from twenty open-source projects. Three human raters independently label each of the 2,000 code review comments as *positive*, *negative* or *neutral* in accordance with the sentimental polarities they perceive in the comment. The decisive sentiment of a particular comment is determined based on majority agreements. Thus, Ahmed et al. produced a three-class dataset consisting of comments with positive, negative, or neutral sentiments. However, in their publicly published dataset, the positive and neutral comments are merged in one *non-negative* class. Therefore, in this work, we have to use this two-class dataset where 19.9% comments express negative sentiments and the rest 80.1% comments convey non-negative sentiments.

III. Sentiment Analysis Tools under Study

We study the following three SE domain specific SA tools released in last couple of years.

SentiStrength-SE: The tool `Sentistrength-SE` [15] is the first domain specific tool especially developed for sentiment analysis in software engineering text. Given piece of text $\mathcal{T}$, `SentiStrength-SE` computes a pair $\langle p_c, n_c \rangle$ of integers, where $+1 \leq p_c \leq +5$ and $-5 \leq n_c \leq -1$. Here, p_c and n_c respectively represent the positive and negative sentimental scores for the given text $\mathcal{T}$. In `Sentistrength-SE`, a given text $\mathcal{T}$ is considered to have positive sentiment if $p_c > +1$. Similarly, a text is held containing negative sentiment when $n_c < -1$. Besides, a text is considered sentimentally neutral when the sentimental scores for the text appear to be $\langle 1, -1 \rangle$.

Senti4SD: The tool `Senti4SD` [6] is a machine learning based tool specifically trained to support sentiment analysis in software engineering related text. By exploiting a suite of both lexicon and keyword-based features, it can detect positive, negative, and neutral sentiments in text. The authors of the classifier claim that it reduces the misclassifications of neutral and positive posts.

EmoTxt: The tool `EmoTxt` [7] is an open-source toolkit that can detect a set of six basic emotions, namely *love, joy, anger, sadness, fear*, and *surprise* from technical text. To convert the emotional expressions to sentimental polarities we use the same procedure as applied to the JIC dataset as described earlier in Section II-A. However, `EmoTxt` cannot identify the polarities of *surprise* expression. To mitigate this issue, from all the datasets, we exclude those comments, which are identified to convey only *surprise* expression by `EmoTxt` (elaborated later in Section IV-A).

IV. Evaluation and Findings

To address the first research question (as mentioned in Section I), we perform a two-stage analysis of comparative accuracies of the sentiment analysis tools. The second research question is addressed through an agreement analysis, as described in Section IV-B.

A. Comparative Accuracy Analysis

The accuracy of sentiment detection is measured in terms of *precision*, *recall*, and *F-score* separately computed for each of the three sentimental polarities (i.e., positivity, negativity and neutrality). Given a set $\mathcal{S}$ of textual contents, *precision* (p), *recall* (r), and *F-score* $(\beth)$ for a particular sentimental polarity e is calculated as follows:

$$p = \frac{|\ \mathcal{S}_e \cap \mathcal{S}_e^t\ |}{|\ \mathcal{S}_e^t\ |}, \quad r = \frac{|\ \mathcal{S}_e \cap \mathcal{S}_e^t\ |}{|\ \mathcal{S}_e\ |}, \quad \beth = \frac{2 \times p \times r}{p + r}$$

where $\mathcal{S}_e$ represents the set of texts having sentimental polarity e (according to ground-truth), and $\mathcal{S}_e^t$ denotes the set of texts that tool t detects to have the sentimental polarity e.

Stage-1 Evaluation: We separately operate the three tools (`SentiStrength-SE`, `Senti4SD`, and `EmoTxt`) on the

JIC (JIRA Issue Comments) dataset and the SOP (Stack Overflow Posts) dataset. We find 302 comments in JIC dataset and 282 posts in the SOP dataset, for which `EmoTxt` finds *surprise* expression only and cannot proceed further to determine polarities of those *surprise* expression. These comments and posts are excluded from the respective datasets to maintain a level-playing field for all the tools.

For each of the three sentimental polarities (i.e., positivity, negativity, and neutrality), we compare the tools' outcome with the respective ground-truth and separately compute precision, recall, and F-score for all the tools for both JIC and SOP datasets. Table II presents the accuracies (in precision, recall, and F-score) of the of the three tools in their detection of *positive*, *negative* and *neutral* sentiments. The average overall accuracies are presented at the bottom three rows. The highest metric values are highlighted in bold.

TABLE II
TOOLS' ACCURACIES FOR JIC DATASET AND FOR SOP DATASET

Dataset	Senti.	Met.	SSE*	Senti4SD	EmoTxt
JIRA Issue Comments	Pos	p	**64.73%**	54.04%	61.26%
		r	**94.29%**	75.20%	70.12%
		$\exists$	**76.76%**	62.89%	65.39%
	Neg	p	**70.96%**	54.78%	34.34%
		r	**78.14%**	41.56%	61.74%
		$\exists$	**70.91%**	47.26%	44.13%
	Neu	p	**91.87%**	80.85%	88.92%
		r	**79.63%**	74.49%	65.98%
		$\exists$	**85.31%**	77.54%	75.75%
Stack Overflow Posts	Pos	p	82.69%	**97.44%**	88.57%
		r	94.15%	**97.44%**	94.15%
		$\exists$	88.05%	**97.44%**	91.27%
	Neg	p	73.45%	**93.03%**	64.93%
		r	78.17%	95.94%	**96.02%**
		$\exists$	75.74%	**94.46%**	77.47%
	Neu	p	80.73%	**97.29%**	94.84%
		r	69.10%	**94.78%**	51.15%
		$\exists$	74.47%	**96.02%**	66.46%
Overall average accuracy		p	77.41%	**79.57%**	72.14%
		r	**82.24%**	79.90%	73.19%
		$\exists$	**79.75%**	79.73%	72.66%

SSE* = SentiStrength-SE

As seen in Table II, the accuracy of `EmoTxt` has remains lower than that of `SentiStrength-SE` and `Senti4SD` for both the datasets. `SentiStrength-SE` achieves the highest accuracy for the JIC dataset while `Senti4SD` achieves the highest accuracy for the SOP dataset. The overall average accuracies also indicate that both `SentiStrength-SE` and `Senti4SD` perform better than `EmoTxt` by a considerable margin. Although it is difficult to distinguish a clear winner, `SentiStrength-SE` can be held superior to `Senti4SD` due to its overall higher recall and slightly higher F-score.

It is interesting to observe that `Senti4SD` and the SOP dataset are from the same authors. Similarly, the JIC dataset is the same dataset on which `SentiStrength-SE` was originally evaluated at the time of its release. Hence, there is a chance of bias and it is worth evaluating all these tools using a dataset on which none of the tools are ever been tested. We carry out such an evaluation in stage-2 using the third dataset described earlier in Section II-C.

Stage-2 Evaluation: We separately operate all the three tools on the CRC (Code Review Comments) dataset. Again, we find 188 comments, which `EmoTxt` identified to express *surprise* emotion only. Similar to the stage-1 evaluation, we exclude these 188 comments from our analysis. Then for non-negative and negative sentimental texts, we separately compute precision, recall, and F-score for all the three tools. The computed accuracy measurements are presented in Table III.

TABLE III
TOOLS' ACCURACIES FOR CODE REVIEW COMMENTS DATASET

Dataset	Senti.	Met.	SSE*	Senti4SD	EmoTxt
Code Review Comments	Non-neg	p	**83.64%**	81.69%	81.45%
		r	92.67%	**93.13%**	82.42%
		$\exists$	**87.92%**	87.03%	81.93%
	Neg	p	50.23%	55.09%	**84.95%**
		r	**34.69%**	28.75%	24.69%
		$\exists$	**41.04%**	37.78%	38.26%
Overall average accuracy		p	66.94%	68.39%	**83.20%**
		r	**63.68%**	60.94%	53.56%
		$\exists$	**65.26%**	64.45%	65.16%

SSE* = SentiStrength-SE

As seen in Table III, all the tools appear to have performed much better in the detection of non-negative sentiments compared to their accuracies in the detection of negative sentiments. `EmoTxt` achieves the highest precision in detecting negative sentiments, `Senti4SD` has the highest recall in the detection of non-negative sentiments. On the other hand, `SentiStrength-SE` achieves the higher precision and F-score for non-negative sentences as well as the highest recall and F-score for negative sentiments.

The overall average accuracies suggest that `EmoTxt` has the highest precision but the lowest recall and the differences from those the other tools are substantial. On the contrary, `SentiStrength-SE` achieves the higher recall and F-score but the lowest precision. The overall average precision of `Senti4SD` is slightly higher than that of `SentiStrength-SE`, but `Senti4SD`'s recall and F-score are lower by small margin than those of `SentiStrength-SE`. Thus, similar to the result of stage-1 evaluation, `SentiStrength-SE` can be considered to have achieved slightly better accuracies than the other tools, in terms of recall and F-score, although the differences can be perceived negligible.

Based on our observations and analyses of results in both stage-1 and stage-2 evaluations, we now derive the answer to the first research question (RQ1) as follows.

Ans. to RQ1: *Accuracies of the tools vary across datasets and sentiments. None of the tools stand out as substantially superior to the other tools. However, `SentiStrength-SE` consistently achieves the highest recall with competitive precision across sentiments and datasets.*

B. Analysis of Agreements

For addressing the second research question (RQ2), we perform an agreement analysis over the tools sentiment detection

results. We compute P^e_{xy} denoting the agreement between tool x and tool y for a particular sentiment e as follows:

$$P^e_{xy} = \frac{\mid \mathcal{S}^x_e \cap \mathcal{S}^y_e \mid}{\mid \mathcal{S}_e \mid} * 100$$

TABLE IV
AGREEMENTS BETWEEN TOOL-PAIRS IN THE DETECTION OF SENTIMENTS

Dataset	Sentiment	SSE* vs. Senti4SD	SSE* vs. EmoTxt	Senti4SD vs. EmoTxt
JIRA Issue Comments	Pos	**77.88%**	72.17%	63.51%
	Neg	51.32%	**72.03%**	49.74%
	Neu	**81.86%**	72.29%	70.00%
Stack Overflow Posts	Pos	**94.35%**	92.31%	93.63%
	Neg	79.02%	80.71%	**93.49%**
	Neu	**82.32%**	83.22%	81.26%
Code Review Comments	Non-neg	**94.70%**	86.05%	83.81%
	Neg	65.00%	66.56%	**66.88%**

SSE* = SentiStrength-SE

The computed agreements between each *pair* of the tools are presented in Table IV. It can be observed that the agreements between the tools vary across different datasets and sentiments. For example, the tools `SentiStrength-SE` and `Senti4SD` show the highest agreement (94.70%) for non-negative sentiments in the CRC (Code Review Comments) dataset whereas the lowest agreement (49.74%) is found between `EmoTxt` and `Senti4SD` in the detection of negative sentiments in the JIC (JIRA Issue Comments) dataset.

We observe two patterns in the agreements of the tools presented in Table IV. First, `SentiStrength-SE` and `Senti4SD` always achieve the highest agreement for non-negative (i.e., positive and neutral) sentiments. Second, for each pair of tools, on every dataset, the lowest agreement is found in the detection of negative sentiments. The only exception to this holds for `Senti4SD` and `EmoTxt` in the SOP (Stack Overflow Posts) dataset.

TABLE V
AGREEMENTS AMONG TOOL-TRIO IN THE DETECTION OF SENTIMENTS

Dataset	Sentiment	Fleiss' κ	Agreement Strength	Reason of Interpretation
JIRA Issue Comments	Pos	0.108	poor	$0.00 \leq \kappa \leq 0.19$
	Neg	0.087	poor	$0.00 \leq \kappa \leq 0.19$
	Neu	0.101	poor	$0.00 \leq \kappa \leq 0.19$
Stack Overflow Posts	Pos	0.498	moderate	$0.40 \leq \kappa \leq 0.59$
	Neg	0.164	poor	$0.00 \leq \kappa \leq 0.19$
	Neu	0.731	substantial	$0.60 \leq \kappa \leq 0.79$
Code Review Comments	Non-neg	0.462	moderate	$0.40 \leq \kappa \leq 0.59$
	Neg	0.265	fair	$0.20 \leq \kappa \leq 0.39$

To further examine the agreements in the tool-trio (i.e., among the three tools), we compute Fleiss' kappa [8] (adaptation of Cohen's kappa for three or more participants), denoted as κ, for each sentiment in all the three datasets. The computed Fleiss' kappa (κ) values and their interpretations are presented in Table V. As seen in the table, there are only one instances of each 'fair' and 'substantial' agreements, two instances of 'moderate' agreements and in all other cases, there are 'poor' agreements among the tool-trio. The tools agree the least in the JIC dataset.

In every dataset, the lowest Fleiss' kappa (κ) values are found for the negative sentiments, again indicating the least agreements among the tools as is also found in the results of tool-pair agreements (Table IV). This can be related to our observations in both Table II and Table III, where, for all the tools, the accuracy of detecting negative sentiments are found consistently lower compared to non-negative sentiments. Hence, we suspect that all the three tools struggle more or less in accurately detecting especially the negative sentiments in text.

Based on our analyses and observations, we now derive the answer to the second research question (RQ2) as follows:

Ans. to RQ2: *Agreements between the tools in detecting sentiments vary largely across different datasets and sentiments ranging between 49.74% and 94.70%. Much of the disagreements among the tools are attributed to their disagreements in the detection of negative sentiments in text.*

V. THREATS TO VALIDITY

It may be argued that the datasets used in this study are not large enough covering all possible categories technical text relevant to software engineering. However, this study includes all the *publicly available software engineering domain specific* sentiment analysis datasets and tools. However, our study does not include `SentiCR` [4], which is another recently released sentiment analysis tool. We deliberately exclude it since the authors of `SentiCR` declared the scope of this tool limited to code review comments only, and thus it could be unfair to evaluate its performance on datasets including JIRA issue comments or Stack Overflow posts.

Blaz and Becker [5] and Ortu et al. [21] developed tools for sentiment analysis in software engineering related text. The tool and dataset of Blaz and Becker [5] are meant for "Brazilian Portuguese" language and thus not comparable with the datasets and tools used in this study. The tools and datasets of Blaz and Becker [5] and Ortu et al. [21] are not publicly available, which is another reason for excluding them from this study.

VI. RELATED WORK

We characterize all the SA tool comparison work including ours in four categories: (1) DI-DI: Comparison of domain independent (DI) tools using DI datasets, (2) DI-DS: Comparison of DI tools using SE domain specific (DS) datasets, (3) M-DS: Comparison of mixed (i.e., DI and DS) tools using DS datasets, and (4) DS-DS: Comparison of DS tools using DS datasets.

(1) DI-DI: Abbasi et al. [3] performed a comparison of *domain independent* sentiment analysis (SA) tools by operating them on five different Twitter datasets. Ribeiro et al. [24] conducted a comparison of 24 unsupervised off-the-shelf sentiment analysis methods. Their evaluation was based on labeled datasets including messages posted on social networks, movie and product reviews, as well as opinions and comments in news articles. In an earlier work Gonalves [9] compared eight sentence-level *domain independent* sentiment analysis methods using a single public DI dataset.

(2) DI-DS: Jongeling et al. [16] compared four *domain independent* SA tools on software engineering dataset and expressed the need for a domain specific SA tool for software engineering text. Islam and Zibran developed `SentiStrength-SE` [15], which is the first software engineering domain specific SA tool (introduced in Section III). In the evaluation, they compared `SentiStrength-SE` against a domain independent tool only. In a later study, Islam and Zibran [14] compared four general purpose SA dictionaries using the JIC dataset introduced in Section II-A.

Recently, Ahmed et al. [4] evaluated seven *domain independent* SA techniques (i.e., *AFINN* [20], *NLTK* [10], *SentiStrength* [25], *TextBlog* [18], *USent* [23], *VADER* [11] and *Vivekn* [19]) using the CRC dataset (Section II-C).

(3) M-DS: Calefato et al. [6], the authors of `Senti4SD` (introduced in Section III), compared their SE domain specific SA tool with *domain specific* `SentiStrength-SE` [15] and *domain independent* `SentiStrength` and `Senti4SD` using the CRC dataset (introduced in Section II-C).

(4) DS-DS: Unlike all the aforementioned work, ours is the first study that compares multiple SE *domain specific* SA tools using multiple publicly available SE *domain specific* datasets. Thus, this work makes a unique contribution to the literature.

VII. Conclusion

In this paper, we have presented the first comparative study of publicly available three software engineering (SE) domain specific sentiment analysis (SA) tools (i.e., `SentiStrength-SE`, `Senti4SD`, and `EmoTxt`) using three SE domain specific datasets.

Our study reveals that the individual tools exhibit their best performance on the dataset they were originally tested at the time of their release. The overall accuracies of the tools tend to decrease when they are operated on a different dataset. The accuracies of the tools largely vary across different datasets and sentimental polarities. Thus, *none* of the tools demonstrates *substantially* superior accuracies across sentiments and datasets. However, `SentiStrength-SE` is found to have consistently exhibited the highest recall and F-score while maintaining competitive precision across all the datasets and sentiments.

From agreement analysis among the tools, we find that the tools' agreements largely vary (between 49.74% and 94.70%) depending on the datasets they are operated on and the sentimental polarities they detect. The tools' agreements remain the lowest in the detection of negative sentiments. Their accuracy also remain lower in the detection of negative sentiments compared to non-negative sentiments. Thus, we suspect that all the tools more or less struggle in accurately detecting negative sentiments in SE text.

We plan to extend this work with new datasets and including an in-depth qualitative analysis to investigate why and in which cases the tools are in disagreements or incorrect in detecting sentiments, especially in the cases of negative ones.

Acknowledgement

This work is supported in part by the SCoRe grant at the University of New Orleans.

References

[1] *Natural Language Toolkit*. http://www.nltk.org/api/nltk.sentiment.html, verified: Jan 2018.

[2] *Stanford Core NLP*. http://stanfordnlp.github.io/CoreNLP/sentiment.html, verified: Jan 2018.

[3] A. Abbasi, A. Hassan, and M. Dhar. Benchmarking twitter sentiment analysis tools. In *LREC*, pages 823–829, 2014.

[4] T. Ahmed, A. Bosu, A. Iqbal, and S. Rahimi. Senticr: a customized sentiment analysis tool for code review interactions. In *ASE*, pages 106–111, 2017.

[5] C. Blaz and K. Becker. Sentiment analysis in tickets for it support. In *MSR*, pages 235–246, 2016.

[6] F. Calefato, F. Lanubile, F. Maiorano, and N. Novielli. Sentiment polarity detection for software development. *Empirical Software Engineering*, pages 1–31, 2017.

[7] F. Calefato, F. Lanubile, and N. Novielli. EmoTxt: A toolkit for emotion recognition from text. In *ACII*, 2017.

[8] J. Fleiss. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378, 1971.

[9] P. Gonçalves, M. Araújo, F. Benevenuto, and M. Cha. Comparing and combining sentiment analysis methods. In *COSN*, pages 27–38, 2013.

[10] M. Hu and B. Liu. Mining and summarizing customer reviews. In *KDD*, pages 168–177, 2004.

[11] C. Hutto and E. Gilbert. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *WSM*, pages 216–225, 2014.

[12] M. Islam and M. Zibran. Exploration and exploitation of developers' sentimental variations in software engineering. *Internation Journal of Software Innovation*, 4(4):35–55, 2016.

[13] M. Islam and M. Zibran. Towards understanding and exploiting developers' emotional variations in software engineering. In *SERA*, pages 185–192, 2016.

[14] M. Islam and M. Zibran. A comparison of dictionary building methods for sentiment analysis in software engineering text. In *ESEM*, pages 478–479, 2017.

[15] M. Islam and M. Zibran. Leveraging automated sentiment analysis in software engineering. In *MSR*, pages 203–214, 2017.

[16] R. Jongeling, S. Datta, and A. Serebrenik. Choosing your weapons: On sentiment analysis tools for software engineering research. In *ICSME*, pages 531–535, 2015.

[17] R. Jongeling, P. Sarkar, S. Datta, and A. Serebrenik. On negative results when using sentiment analysis tools for software engineering research. *Empirical Software Engineering*, pages 1–42, 2017.

[18] S. Loria. *Textblob: Simplified text processing*. Secondary TextBlob: Simplified Text Processing, 2014.

[19] V. Narayanan, I. Arora, and A. Bhatia. Fast and accurate sentiment classification using an enhanced naive bayes model. In *IDEAL*, pages 194–201, 2013.

[20] F. Nielsen. A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. In *MSM*, 2011.

[21] M. Ortu, B. Adams, G. Destefanis, P. Tourani, M. Marchesi, and R. Tonelli. Are bullies more productive? Empirical study of affectiveness vs. issue fixing time. In *MSR*, pages 303–313, 2015.

[22] M. Ortu, A. Murgia, G. Destefanis, P. Tourani, R. Tonelli, M. Marchesi, and B. Adams. The emotional side of software developers in JIRA. In *MSR*, pages 480–483, 2016.

[23] N. Pappas, G. Katsimpras, and E. Stamatatos. Distinguishing the popularity between topics: A system for up-to-date opinion retrieval and mining in the web. In *CLITP*, pages 197–209, 2013.

[24] F. Ribeiro, M. Araújo, P. Gonçalves, M. Gonçalves, and G. Benevenuto. SentiBench - A benchmark comparison of state-of-the-practice sentiment analysis methods. *EPJ Data Science*, 5(23):Open access, 2016.

[25] M. Thelwall, K. Buckley, and G. Paltoglou. Sentiment strength detection for the social web. *Journal of the American Society for Info. Science and Tech.*, 63(1):163–173, 2012.

[26] P. Tourani, Y. Jiang, and B. Adams. Monitoring sentiment in open source mailing lists – exploratory study on the apache ecosystem. In *CASCON*, pages 34–44, 2014.