

On the Subjectivity of Emotions in Software Projects: How Reliable are Pre-Labeled Data Sets for Sentiment Analysis?

Marc Herrmann^a, Martin Obaidi^b, Larissa Chazette^c, Jil Klünder^d

Leibniz University Hannover
Software Engineering Group
Hannover, Germany

^amarc.herrmann@stud.uni-hannover.de

^bmartin.obaidi@inf.uni-hannover.de

^clarissa.chazette@inf.uni-hannover.de

^djil.kluender@inf.uni-hannover.de

Abstract

Social aspects of software projects become increasingly important for research and practice. Different approaches analyze the sentiment of a development team, ranging from simply asking the team to so-called sentiment analysis on text-based communication. These sentiment analysis tools are trained using pre-labeled data sets from different sources, including GitHub and Stack Overflow.

In this paper, we investigate if the labels of the statements in the data sets coincide with the perception of potential members of a software project team. Based on an international survey, we compare the median perception of 94 participants with the pre-labeled data sets as well as every single participant's agreement with the predefined labels. Our results point to three remarkable findings: (1) Although the median values coincide with the predefined labels of the data sets in 62.5% of the cases, we observe a huge difference between the single participant's ratings and the labels; (2) there is not a single participant who totally agrees with the predefined labels; and (3) the data set whose labels are based on guidelines performs better than the ad hoc labeled data set.

© 2022 Published by Elsevier Ltd.

Keywords: Sentiment analysis, software projects, polarity, development team, communication

1. Introduction

The days of one-man software development are long gone [1]. Today, we face software projects getting steadily bigger and more complex, requiring social efforts to work together in development teams such as adequate interaction and communication [1, 2, 3, 4]. Inadequate and insufficient communication and interactions have been proven to be able to harm the productiveness and problem-solving abilities of developers [5, 6]. This, in turn, has a negative influence on the development team as a whole and the software project. As a result, the research field of sentiment analysis (sometimes also referred to as emotion mining) in software engineering has grown steadily over the past few years [7]. Nowadays, there exist multiple tools using different techniques to apply sentiment analysis to data sources from software engineering, including *JIRA* [8], *GitHub* [9], and *Stack Overflow* [10]. A recent systematic literature study [7] shows that almost all of these tools use some form of machine learning, requiring some kind of training data to allow the algorithm to learn which statements, comments, and texts evoke which sentiment polarity, and to be able to measure the performance of these tools.

For this reason, there is a necessity for pre-labeled data sets to develop, train, and test such tools performing sentiment analysis in software engineering. However, manual annotation of data is very time-consuming if one wants to obtain a sufficiently large data set, and the process of labeling data is a difficult task for humans [11, 12]. One possible explanation for this difficulty is the subjectivity of the answers. Consequently, the results may not always be reliable (and they might not represent the perception of developers in general), especially, if the data points are only labeled by a single rater, even when doing so according to specified guidelines. Consequently, multiple raters per data point would be beneficial to solve this issue. Murgia et al. [13] state that already two independent raters labeling each

data point suffice to create a reliable data set, with more than two raters per data point not having a remarkable impact on the measured degree of agreement between the raters.

However, given the diversity of team members in software projects, including their culture, personality traits, and habits, it appears to be unlikely that gold standard data sets adequately reflect all these differences in the team. Consequently, we want to analyze how far gold standard data sets reflect the median perception of software developers as well as, more concretely, the perception of single developers. That is, would a possible team member of a software project assign the same label to a given statement as the gold standard data set? These data sets are used to train sentiment analysis tools that are meant to predict the perception of a developer or a development team (e.g., [9, 10, 14]). Thus, in case of a data set that does not reflect the perception of a developer, it is unlikely that a tool trained on this data set adequately and correctly predicts his or her perception.

In a survey, we asked 180 participants (94 of which we considered in our final population) to rate different statements based on the perceived sentiment polarity raised when reading such statement. We did not provide any guidelines, but used excerpts from two pre-labeled data sets (from the GitHub gold standard provided by Novielli et al. [15] and an ad hoc labeled Stack Overflow data set by Lin et al. [16]).

In this paper, we investigate the overall agreement between a group of raters (represented by the study participants) and the given labels, as well as the agreement between every single rater and the labels. This allows us to draw conclusions on the reliability of the pre-labeled data sets by means of which kind of team member they match (given the nature of subjective statements, it is almost impossible that the labeled data sets match every developer).

Our results point to three interesting findings: (1) Although the median values coincide with the predefined labels of the data sets in 62.5% of the cases, we observe a huge difference between the single participant's ratings and the labels; (2) there is not a single participant who totally agrees with the predefined labels; and (3) the data set whose labels are based on guidelines performs better than the ad hoc labeled data set.

Outline. The rest of the paper is structured as follows: In Section 2, we present related research. Section 3 presents the research design. We present our results in Section 4, which we discuss in Section 5. Section 6 concludes the paper.

2. Background and Related Work

In this section, we will first explain the emotion model that was the basis for the labeling process of one of the data sets used in the remainder of this paper. Afterwards, we discuss different data sets and their annotation processes that were built in the context of software engineering. We end this section with presenting related work.

2.1. Emotion models

Shaver et al. [17] asked psychology students to perform a hierarchical sorting of emotions. This resulted in three levels in the emotion hierarchy. The first level distinguishes between positive and negative emotions. In the second level, five basic emotion categories were used: love, joy, anger, sadness, fear. The sixth emotion surprise, according to Shaver et al. [17], was not considered a basic category because it rarely occurred in the studies, but was nevertheless considered for further analysis. In the third level, emotions are divided into subcategories (for example, sadness is divided into, e.g., guilt or agony). Parrott [18] expanded Shaver et al.'s framework [17] with further details and also distinguishes between three levels: primary emotion, secondary emotion, and tertiary emotion. With the addition of surprise, the primary emotions are equal to the second level of Shaver et al.'s model [17]. The tertiary emotion is a more fine-grained distinction compared to Shaver et al. [17]. These models have been used to label data sets used for sentiment analysis in software engineering.

2.2. Sentiment analysis and data sets in software engineering

According to a recent literature review by Obaidi and Klünder [7], sentiment analysis has been in the focus of several researchers. For example, Novielli et al. [19] describe how they developed a gold standard data set to support the ongoing research on sentiment analysis in software engineering. They used annotation guidelines based on the framework by Shaver et al. [17] to formalize the process of labeling statements from Stack Overflow. They considered the six emotions of the second level, without the subdivision of the third level.

The data set from Novielli et al. [19] contains 4800 data points with each data point representing a question, an answer, or a comment from Stack Overflow. Each data point was assigned a label according to the guidelines by

Shaver et al. [17] independently by three computer scientist student using majority voting to receive the final sentiment label for the data set. The Fleiss' κ [20] values ranged from 0.30 to 0.66. However, since the observed agreement (i.e., the percentage of cases for which raters provided the same annotation) ranged from 0.87 to 0.98, they consider their data set being reliable. The rather small values of Fleiss' κ can be explained by the remarkable chance of random agreement [19].

In a follow-up paper, Novielli et al. [15] created an even larger gold standard data set from GitHub pull-request and commit comments. They used the same methodology and guidelines by Shaver et al. [17] with three raters independently assigning labels to each data point. This time, Novielli et al. [15] mapped the affects from the framework [17] into the three sentiment polarity classes with love and joy being *positive*, and sadness, anger, and fear being *negative*. The “surprise” category was discarded because this emotion could not be transferred adequately [15]. All other data points retrieved the class *neutral* because of the absence of explicit emotions. In total, the authors obtained 7113 data points classified into *positive*, *negative*, and *neutral* [15]. This is one of the largest pre-labeled data set for sentiment analysis in the software engineering domain that is published¹.

Murgia et al. [13], analogously to Novielli et al. [15, 19?], annotated a data set consisting of JIRA issue comments using an emotion model. They used the six primary emotions of level one from Parrott's framework [18], which consists of the framework by Shaver et al. [17] as mentioned before. Among other things, they found that more than two raters do not seem to make a significant difference in terms of agreement. For this they compared Cohen's κ [21] for two raters with Fleiss' κ [20] for more than 2 raters.

Lin et al. [16] analyze the limits of sentiment analysis in the software engineering domain. For this purpose, they manually labeled sentences extracted from Stack Overflow. As there is no mention of the use of some kind of guidelines or framework to assign the sentiment polarity labels to the data set, we presume that ad hoc labeling was used. Zhang et al. [22] use the data set by Lin et al. [16] in their replication study. They tested various software engineering specific sentiment analysis tools and pre-trained transformer-based models on six different data sets from the software engineering domain [22].

Ahmed et al. [9] build a data set, which includes negative and non-negative documents. Therefore it is a binary-class data set. The documents consists of multiple comments. They collected them from 20 open-source projects that practice tool-based code reviews supported by the same tool (e.g., Gerrit). Three of the authors labeled the sentences based on their perception as recipients of the message. Thus, the evaluation was probably ad hoc.

However, all these papers stated that after discussing or majority voting, they resolved the disagreement cases and thus built a data set for evaluation, without having these data sets validated by external computer scientists.

2.3. Application of sentiment analysis in software engineering

Obaidi and Klünder [7] examined the development and application of sentiment analysis in software engineering through a systematic literature review ($n = 80$ papers). Among other results, their findings show that both GitHub and Stack Overflow are among the top three data sources used for sentiment analysis. In addition, domain independent sentiment analysis tools have been found to often leading to poor performance in the software engineering domain, because of certain terms being used differently in software engineering than elsewhere (e.g., “to kill a process”) [7]. Those two findings reinforce the value of the data sets by Novielli et al. [15] and Lin et al. [16]. However, Obaidi and Klünder [7] also found that a common difficulty for authors lies in the subjectivity of labeling a statement with a sentiment alone, meaning that different people would already assign different labels to the same statements [22, 23, 24, 25]. Imtiaz et al. [23] for example annotated GitHub comments based on the annotation guide provided by Mohammad [26]. A total of 3 coders were involved in the sentiment annotation process, but their Cohen's κ values ranged from 0.27 to 0.38. They suggested that more coders could ensure more reliable human evaluation.

This commonly faced issue motivated the research on the subjectivity of sentiment perception in this paper.

In this paper, we compare the sentiment labels assigned by different raters from the gold standard data set by Novielli et al. [15] and the ad hoc annotated data set from Lin et al. [16] with the perception of possible software project team members (without using any guidelines) in the software engineering domain. To achieve this, we analyzed labeling data from 94 participants in a survey and compare the participants' results to the excerpts of two publicly available pre-labeled data sets [15, 16].

¹The GitHub gold standard data set is available on Figshare.

3. Survey Design

In the following subsections, we present our research objective and the research questions, as well as the structure and our ideas behind the creation of the survey.

3.1. Research Objective and Research Questions

Our main research objective is to compare the perceptions of software project team members with the polarity provided by pre-labeled data sets for sentiment analysis. This helps improve the reliability and accuracy of predictions for sentiment analysis tools in software engineering domains. To achieve this goal, we pose the following research questions:

RQ1: How do the median labels assigned by the study participants differ from the predefined labels?

RQ2: How much do each participant’s labels differ from the predefined labels?

RQ3: How do the results differ between ad hoc and guideline-based labeled data sets?

3.2. Instrument Development

We used the survey method [27] to collect our data, implemented as an online questionnaire.

3.2.1. Survey Structure

The final questionnaire consisted of five blocks of questions (number of questions in parentheses): Demographics (4), affiliation to computer science (2), programming experience (5), labeling (1 repeated for 100 statements), criteria (1). A detailed overview of the survey structure is presented in Table 1.

In the beginning, we collected demographic data such as age, gender, if English is the native language, and the frequency of communication on English. The next block of questions asked whether the participants identify as computer scientists, and what their current professional status is. Next, the questions dealt with the programming skills, the experience in professional work environments and with team work.

Thereafter, the survey had a block of 100 statements. The 100 statements consisted of 48 different statements from the gold standard data set [15], and 48 statements from the ad hoc labeled data set [16], and 4 duplicates. We added 4 random statements (2 *positive*, 1 *neutral*, and 1 *negative*) from the GitHub gold standard data set [15] twice to make total of 100 statements. This allows to check how consistent the participants labeled the statement, i.e. to recognize if they choose two different labels for any of the duplicate statements during the survey. The participants were asked to classify each statement as one of the three polarity classes *positive*, *neutral*, and *negative* (without any given guidelines). The selection of these statements is presented in more detail in the next subsection. The 100 statements in the survey were randomly selected in a way leading to 32 *positive*, 32 *neutral*, and 32 *negative* statements (plus 4 duplicates), making a nearly perfect one third split for each of the three sentiment polarity classes². The statements were put together in blocks of ten statements. Both the ten blocks and the ten statements in each block were randomly ordered.

The last question asked the participants to explain how they selected the labels by presenting predefined answer options such as having focused on the statement content or the statement tone, as well as a free-text answer.

3.2.2. Selection of Reference Data Sets

Answering the research questions requires a comparison between the data provided by our survey and already existing data sets. Consequently, we wanted the participants to label statements from data sets that are established when training sentiment analysis tools in software engineering. We selected two different data sets (and thereby two different platform sources) with one being guideline-based annotated and one being ad hoc labeled. The GitHub gold standard data set from Novielli et al. [15] was selected as guideline-based data set and the Stack Overflow data set from Lin et al. [16] was the reference for the ad hoc labeled data set. The GitHub data set [15] is the largest data set (7122 statements) to the best of our knowledge and was built most recently (in 2020). It was used and evaluated in

²Note that we did not tell the participants that the statements are equally distributed among the polarity classes to avoid biases.

Table 1. Survey design

	Questions	Answer Options
Demographics	How old are you?	18 - 99
	What is your gender?	Male
		Female
	Is English your native language?	Yes
		No
	How often do you communicate in English?	Every day
		Multiple times a week
Once a week		
Once in a while		
Never		
Affiliation to CS	Would you identify yourself as computer scientist?	Yes
		No
	What is your professional status?	Student
		Working in Academia
		Working in Industry
		Retired
		Unemployed
Other		
Programming Experience	Do you have any experience with programming?	Yes
		No
	How would you rate your programming skills?	(bad) 1 - 5 (good)
	How many years of professional experience do you have as a developer?	0 - 99
	How familiar are you with working as a developer with a team?	(hardly) 1 - 5 (highly)
	How many years of professional experience do you have in working with a team?	0 - 99
Labeling	How would you label the following sentences regarding its polarity based on your perception? (for 100 statements)	Positive
		Neutral
		Negative
Criteria	What criteria did you use to decide on the polarities of those sentences?	Content
		Tone
		Other

the context of sentiment analysis in software engineering in previous papers (e.g., [15, 22, 28]). The Stack Overflow data set [16] contains 1500 statements and was used also in many papers (e.g., [16, 22, 29]).

From each of the two data sets, we randomly selected 48 statements with 16 *positive*, 16 *negative*, and 16 *neutral* statements, resulting in a total of 96 statements. We evaluate the performance of both data sets separately to compare both guideline-based and ad-hoc annotation to the perception of the participants, as well as both data sets in combination to evaluate the general discrepancies between the labels predefined by the scientific authors and the participants. Please note that, for simplicity, we will refer to these 48 selected statements as the Stack Overflow and GitHub data sets as well as the combined data set (96 statements) in the remainder of the paper.

3.3. Data Collection

The survey was hosted using *LimeSurvey* on the server of the Software Engineering Group at Leibniz University Hannover. We mainly distributed the survey via e-mail. We invited computer science students (BSc and MSc) at Leibniz University Hannover attending our lectures³, as well as computer science doctoral students, postdocs, and research assistants (for which we manually extracted the contact information from the institutional websites). In addition, the survey was sent to publicly available e-mail addresses from researchers who published papers about sentiment analysis in software engineering themselves to ask them to participate and to share the survey with their networks. We extracted this list of authors from a recent systematic literature study [7]. We also shared the survey via social networks such as Twitter, Facebook, LinkedIn, and XING. In the invitation text, we included a short description of the survey, the estimated time of 10 - 20 minutes, and the link to the survey. When distributing the survey, we clearly stated that our target group are computer scientists with programming experience. However, to mitigate the risk of retrieving answers from outside the target group, we inserted three questions asking for (1) the identification as a computer scientist or developer, (2) programming experience, and (3) experiences with software development in teams. The survey was available from the end of April 2021 until the beginning of November 2021. The raw data set is available via Zenodo [30].

3.4. Data Pre-Processing

In total, we received 180 responses. We collected 127 parameters for each data point, some of them resulting from optional questions.

The definition of a subset of the 180 data points being suitable for the analysis presented in this paper was done based on two conditions. As an incomplete data point can still provide interesting insights for answering the research questions, we did not exclude incomplete data points per sé. First, we removed 17 data points that answered either question on programming experience or being a computer scientist (“*Would you identify yourself as a computer scientist (e.g., computer scientist student, developer, etc.)?*” or “*Do you have any experience with programming (e.g programming a software, website, an app, etc.)?*”) with a “No” as our target group were persons who are potential team members in a software project team. As a second criterion, we only included data points where participants had at least annotated one of the statements with their perceived sentiment polarity, removing additional 69 data points. This decision leads to a varying n -value for the single statements for the different analysis steps. Thus, we report the n -value separately for the different analyses. Applying these criteria led to a final data set consisting of 94 data points that were used for the further analyses.

3.5. Data Analysis

We analyzed our collected and pre-processed data using Python with various packages for scientific research including *pandas* [31], *NumPy* [32], *Matplotlib* [33], and *scikit-learn* [34].

³Note that, due to privacy reasons, we were not allowed to invite our students (or students from other universities) via e-mail, but we distributed the information on our survey via the internal software tool used to support lectures. This way, we only contacted students from Leibniz University Hannover, but we asked colleagues from other universities to do the same and to share the survey with their students.

3.5.1. Analysis Procedures for RQ1

To answer our first research question, we calculated the median sentiment class from all the participants' annotations for each statement using the ordinal order *negative* < *neutral* < *positive*. Note that we opted for the median rather than for a majority vote (which is often used incorrectly in the context of sentiment analysis) for the following reason: Assume we have three raters/voters and each of them assigns a different sentiment polarity class to a given statement. The votes are therefore *negative*, *neutral*, and *positive*. The majority vote is ambiguous in this case, as each class received the same amount of votes, and a random sentiment polarity class would be returned with common methods. The median is well defined on the other hand, because we take the ordinal order *negative* < *neutral* < *positive* into account. The sentiment polarity class *neutral* has an even amount of sentiment polarity values above and below it (one each). For this reason we have decided to use the median instead of majority vote for our analysis. We then compared the resulting median sentiment labels with the predefined labels from the ad hoc labeled Stack Overflow data set, the guideline-based labeled GitHub data set, and both data sets in combination. We calculated the absolute and relative counts of coinciding labels (median label vs. predefined label) as well as cases of mild and severe disagreement, as proposed by [15]. Finally, we calculated the observer agreement using Cohen's κ [21]. That is, we considered the following four metrics:

- **Agreement (also referred to as perfect agreement):** Absolute and relative amount of statements that are annotated with the same label both in the original data sets and from the participant(s)
- **Mild Disagreement:** Absolute and relative amount of statements that are annotated with differentiating labels in the original data sets and from the participant(s) with a distance of 1 (i.e. positive - neutral, neutral - negative)
- **Severe Disagreement:** Absolute and relative amount of statements that are annotated with differentiating labels in the original data sets and from the participant(s) with a distance of 2 (i.e. positive - negative)
- **Cohen's κ :** Calculation of the interrater agreement between the predefined labels and the participant(s)

For a more detailed description of the differences between the participants' median labels and the predefined labels, we also calculated precision, recall and F_1 -score for each of the three data sets (from Stack Overflow, from GitHub, and in combination), as well as the confusion matrices. To calculate these values, we assumed the calculated median labels for each statement as the predicted label and used the predefined labels as the true labels.

3.5.2. Analysis Procedures for RQ2

To answer our second research question, we explored our data in more detail. That is, we calculated the agreement and Cohen's κ [21] between each set of individual participant labels and the predefined labels from the two data sets and the combined data set. Note that we only calculated the values if the individual participant had annotated at least 10% of statements from the corresponding data set, as we assumed an amount of less than 10% leading to too few data points to compare so that the results would not be meaningful. The six resulting arrays (agreement and Cohen's κ for the three data sets (from Stack Overflow, from GitHub, and in combination) of agreement and Cohen's κ were analyzed descriptively by calculating minimum, maximum, mean, and standard deviation values.

3.5.3. Analysis Procedures for RQ3

To answer RQ3, we compared the results of the analyses for RQ1 and RQ2. To measure if the difference of participant agreement and κ -values between the two data sets is statistically significant, we applied hypotheses testing. In particular, we tested the main hypothesis H1 presented in Table 2 at a significance level of $p < 0.05$. However, as the main hypothesis asks for a difference between the results for the two data sets in general, we formulated the two sub-hypotheses H1.1 and H1.2. To analyze H1.1 and H1.2, we first tested the respective data for normal distribution using the Shapiro-Wilk test [35]. In case of normal distribution, we used the repeated-measures t-test [36] to test the hypothesis. Otherwise, we opted for the Wilcoxon signed-rank test [37]. As we tested two sub-hypotheses for one main hypothesis H1, we applied the Bonferroni correction [38] leading to an adjusted significance level of $p_{corr} = p/2 = 0.025$. As soon as one of the sub-hypotheses results in a p -value lower than p_{corr} , we consider H1 to be significant. Note that we only considered data points of participants who have labeled at least 10% of statements in both of the two data sets.

Table 2. Null hypotheses to compare the two data sets

H1 ₀	There is no difference between the two data sets by means of the predefined and the participant’s labels.
H1.1 ₀	There is no difference in the agreement between the two data sets comparing the predefined labels and the participant’s labels.
H1.2 ₀	There is no difference in Cohen’s κ between the two data sets comparing the predefined labels and the participant’s labels.

4. Results

We conducted the data cleaning and the data analysis as described in Section 3.4 and Section 3.5. In the following subsections, we present the results of each analysis step.

4.1. Demographics

The participants had an average age of 27.61 years (min: 18 years, max: 55 years, SD: 6.57 years, $n = 93$). Seventy-seven (81.9%) of the participants stated their gender as male and 17 (18.1%) as female ($n = 94$). Only 2 (2.2%) participants were native English speakers, and 91 (97.8%) were foreign language speakers ($n = 93$). However, only 2 (2.2%) participants reported that they never communicate in English (but still considered their English as sufficient to participate in the study), while 30 (33.0%) participants reported on communicating in English once in a while, 13 (14.3%) participants once a week, 20 (22.0%) participants multiple times a week, and 26 (28.6%) participants every day ($n = 91$). For their professional status most of the participants reported being students (66 participants; 70.2%), while 27 (28.7%) participants were working in industry, 15 (16%) participants were working in academia, 3 (3.2%) participants selected the category "Other", and 2 (2.1%) participants were unemployed ($n = 94$, multiple answers were possible). The participants rated their own programming skills on a Likert scale from 1 (beginner) to 5 (expert) with a median of 3 ($n = 94$). On average, participants had 2.957 years of professional experience as a developer (min: 0 years, max: 25 years, SD: 4.674 years, $n = 93$). Considering the familiarity of working as a developer within a team (Likert scale from 1 (no experience) to 5 (very experienced)) the participants stated a median value of 3 ($n = 93$). The years of professional experience as a developer working in a team ranged from 0 to 25 years with an average of 1.957 years (SD: 3.736 years, $n = 92$).

4.2. Comparison Between Median Labels and Predefined Labels

The overall results of agreement, the matches between the median participants perceptions and the predefined labels, and the resulting value of Cohen’s κ are summarized in Table 3. Note that, in this step, we compare the median rating of all participants in the study with the predefined labels of the data set. In the following, we present more detailed results on the whole data set and distinguish between the statements originating from Stack Overflow and GitHub.

4.2.1. Overall Agreement

In total, in 60 out of 96 cases (62.5%) the median rating of the participants coincides with the predefined labels in the two data sets [15, 16]. This agreement results in a Cohen’s κ -value of 0.4375, which can be considered moderate agreement according to the scale provided by Landis and Koch [39].

In case of the ad hoc labeled Stack Overflow data set, for 27 out of 48 statements (56.25%) the median of our participants coincides with the labels given in the data set. This is considered fair agreement with a Cohen’s κ of 0.34375. In the 21 (43.7%) cases where the median label of the participants does not coincide with the labels given by the data set the disagreement is *always* mild, i.e. there are no cases of severe disagreement.

For the 48 statements from the GitHub data set, we retrieve an agreement between our participants and the predefined labels of 68.75% (33 out of 48 statements are assigned the same label). This results in a Cohen’s κ of 0.53125, which is considered moderate agreement. However, there are 7 (14.6%) cases of severe disagreement. Additionally 8 (16.7%) cases of mild disagreement between the median labels of the participants and the predefined labels occurred.

Table 3. Comparison between median participants' and predefined labels

Data set	Size	Agreement	Mild Disagr.	Severe Disagr.	Cohen's κ
Combined	96	60 (62.5%)	29 (30.2%)	7 (7.7%)	0.438
SO	48	27 (56.3%)	21(43.7%)	0 (0.0%)	0.344
GitHub	48	33 (68.8%)	8 (16.7%)	7 (14.6%)	0.531

Finding 1: For the 96 statements, we find a moderate agreement between the participants perceptions and the predefined labels with a Cohen's κ of 0.4375.

Finding 2: For the 48 ad hoc pre-labeled statements, there is a fair agreement ($\kappa = 0.3438$) between the participants and the predefined labels and no cases of severe disagreement.

Finding 3: For the 48 guideline-based labeled statements, the agreement between the participants and the predefined labels is moderate ($\kappa = 0.5313$), but there are 7 cases of severe disagreement (14.6%).

4.2.2. Precision, Recall, and F1

Looking into more details of the (dis-)agreements between the median values of our participants and the predefined labels of the data sets leads to the results presented in Table 4. The table summarizes the precision, recall, and F_1 -score for each data set and each sentiment polarity class.

Table 4. Precision (P), recall (R), and F_1 -score for each data set

	Polarity Class	P	R	F1
<i>Combined data set</i>	Positive	0.770	0.313	0.444
	Neutral	0.527	0.906	0.666
	Negative	0.750	0.656	0.700
<i>Stack Overflow data set (ad hoc)</i>	Positive	0.667	0.250	0.364
	Neutral	0.424	0.875	0.571
	Negative	1.00	0.563	0.720
<i>GitHub data set (guideline-based)</i>	Positive	0.857	0.375	0.522
	Neutral	0.682	0.938	0.789
	Negative	0.632	0.750	0.686

For the whole data set (i.e., the combination of statements from both data sets), we have the best precision (0.77) for the *positive* class, which has the worst recall (0.313), resulting in a F_1 -score of 0.444. The negative class reaches almost the same precision (0.75) and a recall of 0.656. The neutral class has a precision of 0.527 and achieves the highest recall value of 0.906.

The Stack Overflow data set achieves a perfect precision of 1.00 for the negative class, meaning that all negative values assigned by the participants are predefined to be negative by the data set. However, as the participants did not “find” all negative statements (but considered some as neutral), the recall of the negative class is 0.563. The positive class achieves a precision of 0.667 and the lowest recall of 0.25, meaning that several positive statements have not

Table 5. Confusion matrices for the data sets

Predefined Label		Median Label		
		Positive	Neutral	Negative
<i>Combined data set</i>	Positive	10	16	6
	Neutral	2	29	1
	Negative	1	10	21
<i>Stack Overflow data set (ad hoc)</i>	Positive	4	12	0
	Neutral	2	14	0
	Negative	0	7	9
<i>GitHub data set (guideline-based)</i>	Positive	6	4	6
	Neutral	0	15	1
	Negative	1	3	12

been identified as positive. The neutral class has the lowest precision with 0.424 and the highest recall of 0.875, meaning that several statements classified as neutral are positive or negative, and that several neutral statements have been identified as neutral.

The GitHub data set has in total the highest values for precision, recall, and F1-score. The positive class has again the highest precision (0.857), but the lowest recall (0.375). The negative class has a precision of 0.632 and a recall of 0.750. The neutral class has a precision of 0.682 and a recall of 0.938.

Finding 4: The neutral class always has the highest recall, but the worst precision. Thus, our participants were good at identifying the predefined neutral labels as neutral, but in addition classified many other statements as neutral although they had a predefined label of positive or negative.

Finding 5: The positive class often has the lowest recall, but the best precision (except for SO). Thus, most statements labeled as positive by our participants were also predefined to be positive, but a lot of predefined positive statement were instead assigned labels of either neutral or negative by the participants.

4.2.3. Confusion Matrices

The confusion matrices for the whole data set and the predefined labels in the data sets from Stack Overflow and GitHub are shown in Table 5.

In total, we have 32 positive, 32 negative, and 32 neutral statements. In case of the neutral statements, 29 of 32 have been identified as neutral by the participants of our study, but only 21 negative and 10 positive. In these cases, the median labels of our participants coincide with the predefined labels of the data set. The participants identified two neutral and one negative statements as positive. Sixteen positive and 10 negative statements have been identified as neutral. In addition, the participants rated six positive and one neutral statement as negative.

The confusion matrix of the Stack Overflow data set in Table 5 shows that no true positives have been classified as negative from the median participant labels. The same is true for true negative and positive as well as true neutral and negative. In addition, we observe that only 6 statements (12.5%) were labeled as positive, while 33 (68.75%) were labeled as neutral, and 9 (18.75%) were labeled as negative. For the negative sentiment polarity class, only 9 statements were labeled as negative, instead of the authors 16 statements.

For the GitHub data set, true neutral statements are never classified as positive by the median participant labels, each other sentiment polarity class has been misclassified as one of the other two polarity classes at least once. More

so, there are as many correctly classified true positive statements as false negative ones. This is notably interesting as we selected 16 statements from each sentiment polarity class of author labels, making an even split for each sentiment class out of the total 48 statements from each data set. However, we observed a notable discrepancy in the distribution of the median sentiment polarity classes calculated from the participant perceptions. Similarly, we observe that only 7 statements have been labeled as positive, while 22 were labeled as neutral, and 19 were labeled as negative. In both cases, the positive sentiment polarity class was assigned notably less than in the pre-labeled data set (6 and 7 times instead of 16). On the other hand, more statements (19 instead of 16) have been annotated as negative than by the pre-labels.

In both cases, the neutral label was chosen frequently by the participants, given that for the GitHub data set 22 instead of 16 statements were annotated with neutral, and for the Stack Overflow data set more than twice (33 instead of 16) the number of statements were annotated with neutral compared to the predefined labels, making up more than $\frac{2}{3}$ of the overall annotated sentiment labels.

Finding 6: Comparing the median participants' label and the predefined labels, we observe some kind of pessimism by the participants. They rarely assign the positive class to a statement (in particular compared to the negative class).

Finding 7: Considering the confusion matrices, we again observe a better performance of the guideline-based data set compared to the ad hoc labeled one.

4.2.4. Cases of Severe Disagreement

As depicted in Table 3 and Table 5, we observe 7 cases of severe disagreement between the calculated median labels of our participant's and the original labels provided by the authors for the guideline based GitHub data set. Since these cases are arguably the most interesting ones, we investigated the 7 cases manually to try to derive reasons for the statements' controversial nature. Out of the 7 cases of severe disagreement only one statement was labeled negative by the authors and perceived as positive by the participants, for the other six cases the perception was vice versa.

Statement: *“oh nice find, that's been bugging the crap out of me”*

Possible Explanation: We assume that this statement was labeled as negative by the authors because it contains the words “*bugging*” and “*crap*”, which indeed, on their own, are afflicted with a negative sentiment polarity. However, they only occur in the second half of a compound sentence. In the first half, the sentence has a positive sentiment polarity, praising the other person (presumably for finding a bug in the software), and only then explaining his appraisal with the explanation that what the other person has solved annoyed him/her beforehand.

One of the opposing example where the author label was positive and the calculated median label of the participants was negative was the following:

Statement: *“I have no idea either, I just trust the spray guys”*

Possible Explanation: We assume the authors have chosen the label positive because the word “*trust*” was used, which has a positive sentiment polarity. The term “*trust*” is also mentioned in the emotion framework by Shaver et al. [17] under the category “Love” that was used by the authors: “. . . the raters were trained to provide a polarity label based on the emotion detected according to the Shaver model. . .” [15]. The term *spray* used here is slang and means “*To use an automatic weapon to fire blindly and rapidly, releasing a large amount of bullets at one time.*”, according to *Urban Dictionary*. In this sentence the term can be interpreted so that the writer has no idea to why a solution to a bug works, he just trusts randomly attacking the issue with common methods. This is probably a reply to a question as to why the solution given prior fixes the problem. Since the writer has no helpful explanation himself however, the statement can be perceived as negative, which was the case for the participants. Indeed the criterion of helpfulness was also mentioned by the participants as being used for annotating the sentiment polarities, which we discuss in more detail in subsection 4.4.

Overall we observe, that in the cases of severe disagreement in the guideline based GitHub data set, the author labels usually reflect the sentiment polarity emitted by individual words that compose only parts of a compound sentence. The participants on the other hand tried to guess the context of the message reflected in the information given by the statement. In the given examples above we demonstrated, that this can make a significant difference. The annotation criteria used by the participants is described in subsection 4.4 and supports these findings further.

4.3. Match of Participant Perceptions

In a next step, we compared every single participant’s selection of polarity classes with the predefined labels of the data set.

4.3.1. Agreement

Table 6 summarizes the agreement of the participant’s labels in comparison to the predefined labels. For the combined data set, we have a maximum agreement of 0.75 and a minimum of 0.167 (mean: 0.489, SD: 0.137). For the Stack Overflow data set, we have a minimum of just 0.091 and a maximum of 0.833 (mean: 0.47, SD: 0.147). The GitHub data set performs slightly better than Stack Overflow with a minimum of 0.188, a maximum of 0.917, and a mean agreement of 0.517 (SD: 0.158). The distributions of all participants’ agreement values for the Stack Overflow and GitHub data set are shown in Figure 1.

Table 6. Distribution of the calculated agreement values

Data set	min	mean	max	SD	<i>n</i>
Combined	0.167	0.489	0.750	0.137	80
Stack Overflow	0.091	0.470	0.833	0.147	83
GitHub	0.188	0.517	0.917	0.158	82

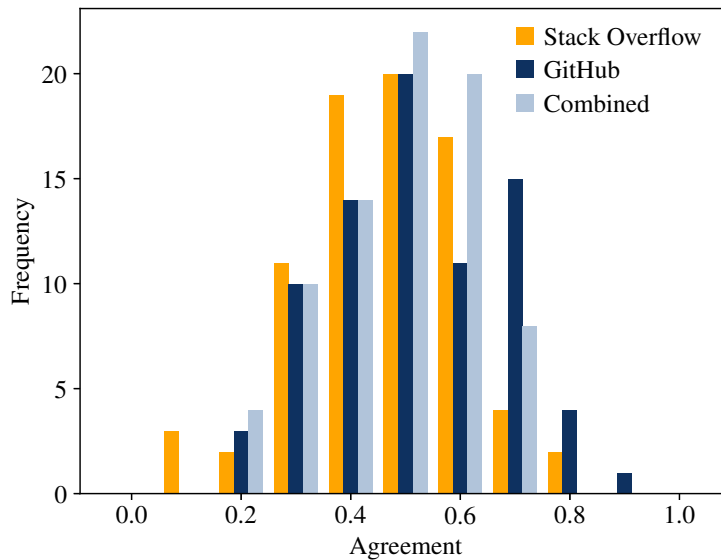


Figure 1. Frequency distribution of agreement ($n = 78$)

4.3.2. Cohen’s Kappa Between the Participants and the Predefined Labels

Table 7 summarizes the agreement between each participant and the predefined labels based on Cohen’s κ . A negative κ -value indicates an agreement lower than by chance [39].

For the combined data set, the κ -values range from -0.253 to 0.627 (mean: 0.235, SD: 0.203).

For the Stack Overflow data set, the κ -values range from -0.314 to 0.75 with a mean of 0.208, which is considered slight agreement [39], and a standard deviation of 0.208 ($n = 83$). Remarkably, 75% of the κ -values are below 0.354, which is only considered fair agreement (50% = 0.219, 25% = 0.093). As with the agreement before, the GitHub data set performs better in terms of Cohen’s κ .

For the GitHub data set, the participants’ κ -values range from -0.219 to 0.875, with an average of 0.275 and a standard deviation of 0.238. The 75% quartile is at 0.5, which is considered moderate agreement [39].

Table 7. Distribution of the calculated κ -values

Data set	min	mean	max	SD	n
Combined	-0.253	0.235	0.627	0.203	80
Stack Overflow	-0.341	0.208	0.750	0.208	83
GitHub	-0.219	0.275	0.875	0.238	82

Figure 2 shows the distribution of the calculated agreement (by means of Cohen’s κ) between each participant and the predefined labels for both data sets (divided into the classes proposed by Landis and Koch [39] to interpret the κ -values). Seemingly, the guideline annotated GitHub data set performs slightly better than the ad hoc labeled Stack Overflow data set.

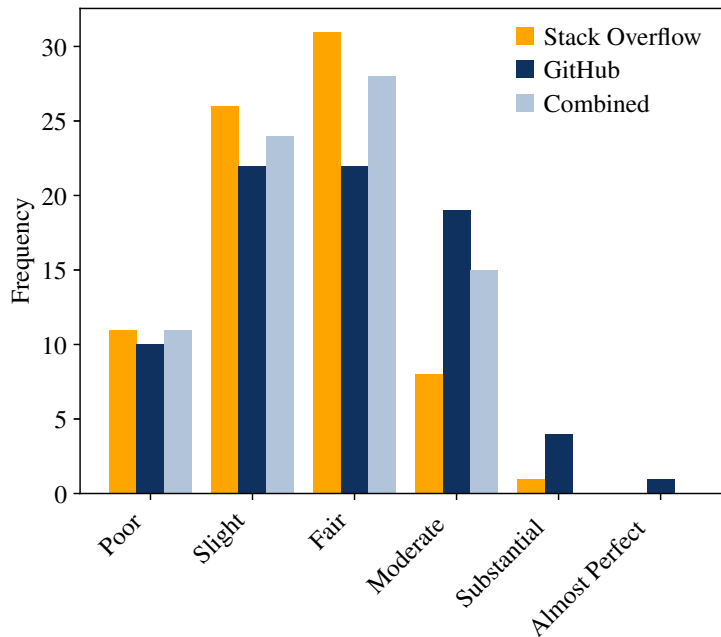


Figure 2. Frequency distribution of agreement strength classes, according to Landis and Koch [39] ($n = 78$)

However, to analyze this difference quantitatively, we tested the hypotheses presented in Table 2. The Shapiro-Wilk test for normal distribution revealed that the agreement and Cohen’s κ are normally distributed for both data sets (cf. Table 8). Therefore, we used the repeated-measures t-test for H1.1 and H1.2.

Table 8. Results of the Shapiro-Wilk test for normal distribution

Data Set	Variable (X)	W	p	$X \sim N$
Stack Overflow	Agreement	0.985	0.499	Yes
GitHub	Agreement	0.985	0.496	Yes
Stack Overflow	Cohen's κ	0.990	0.778	Yes
GitHub	Cohen's κ	0.985	0.504	Yes

The results of the repeated-measures t-test are shown in Table 9, revealing significant differences for both data sets by means of agreement and Cohen's κ . As both p -values are below the significance level of 0.05, we can reject H1.1₀ ($T = -3.395$, $p = 0.00109 < 0.05$) and H1.2₀ ($T = -3.56$, $p = 0.00063 < 0.05$). Summarizing, we can also reject H1₀, as the p -values are below the corrected significance level, and, thus, we can assume that there is a difference between the two data sets with regard to the match between the predefined and the participants' labels.

Table 9. Results of the repeated-measures t-test

Variable	T	p	$p < p_{corr}$
Agreement	-3.395	0.00109	Yes
Cohen's κ	-3.560	0.00063	Yes

Finding 8: There is a significant difference in the agreement of the participants with the predefined labels comparing the guideline-based and the ad hoc labeled data sets.

Finding 9: There is a significant difference in Cohen's κ of the participants with the predefined labels comparing the guideline-based and the ad hoc labeled data sets.

4.4. Annotation Criteria

After labeling the statements, we asked the participants what criteria they used for annotating the statements. Predefined answers included the tone (i.e. the mood implied by the writer's choice of words and the emotions they invoke on the reader), the content (i.e. the information of the statement, is something good or bad described?), and other (text field); multiple answers were possible. We placed this question after the labeling process on purpose, so that participants would label the statements without any influence, where as placing this question beforehand could lead to thinking about the options and choosing one beforehand and sticking with that choice. Out of 62 total responses to this question, 57 participants (91.9%) stated that they used the tone of the statements, while 43 (69.4%) stated that they used the content, and 15 (24.2%) entered custom answers (other) in addition. We evaluated the 15 custom participant answers to this question by hand for a further analysis. Out of the 15 custom answers we found that 6 participants mentioned the use of emoticons (e.g., “:-)”), or other slang (e.g., “LOL”) for determining the sentiment polarity. The constructiveness of a post by being helpful, informative, or giving a solution to a problem was considered by 5 participants. Another 4 participants stated some form of guessing the context of the statement and guessing their own emotional response to it, based on their experience (e.g., “Basically, I tried to guess the tone of the message, reflecting how the person typing the comment or reading it might feel, as opposed to just communicating a technical fact.”). The lack of context was also criticized by one participant as making the annotation task difficult. Despite this fact, the context is still not considered by the sentiment analysis tools.

5. Discussion

We end this paper by discussing our results, answering the research questions, and presenting threats to validity.

5.1. Answer to Research Questions

Based on our results, we can answer the research questions as follows:

RQ1: We observe a huge difference between the median labels of the study participants and the predefined labels. From the 96 statements, the median label of the participants only coincides with the predefined labels in 60 cases, leading to an agreement of 0.625.

RQ2: Looking at every single participant’s perception in comparison to the predefined labels, we again observe a wide variety in the agreement. There are participants who achieve a very high agreement (for the combined data set, the maximum is at 0.75), but others achieve very low values, pointing to substantial disagreement by means of Cohen’s κ .

RQ3: In almost all cases, the participant’s labels coincide more with the labels predefined by the GitHub data set than the Stack Overflow data set. That is, a data set that is labeled using concrete guidelines seems to better reflect the average perception of software project team members than an ad hoc labeled data set. The statistical tests also confirm that there is a significant difference between the two data sets with respect to agreement and Cohen’s κ . However, we also observed a non negligible amount of severe disagreements between the participant’s and authors in case of the guideline-based GitHub dataset, while the participant’s only had mild disagreements with the original labels from the ad hoc labeled Stack Overflow data set.

5.2. Interpretation

Based on the results of our study and the answers to our research questions, we make three remarkable observations:

- (1) Although the median values coincide with the predefined labels of the data sets in 62.5% of the cases, we observe a huge difference between the single participant’s ratings and the labels.
- (2) There is not a single participant who totally agrees with the predefined labels; ranging from barely substantial agreement to substantial disagreement.
- (3) In most cases, the labels of the guideline-based data set coincide more with the median participant(s) labels than the labels of the ad hoc labeled data set.

Besides the threats to validity that may have influenced our results (see Section 5.4), there are other possible explanations for these results.

(1) Both data sets integrated in our study are meant to serve as a training set for sentiment analysis tools. That is, the labels assigned to the statements in the data set shall represent the perception of a “typical”, i.e., of an ordinary member of a development team. Thus, the observation that the median perception of the participants coincides with the predefined labels in the data sets would have been expected.

(2) However, despite the fact that the participants agree on average with the predefined labels, we observe remarkable discrepancies between single participants and the predefined labels. As described in Section 2, it has already been mentioned by Imtiaz et al. [23] that label assignment by human raters without a coding scheme could lead to different understanding of sentiments in the field of software engineering among them. Novielli et al. [15, 40] showed that the absence of clear guidelines for annotation can lead to noisy gold standards data sets. However, our results showed a similar discrepancy between ad hoc labels assigned by our participants compared with both ad hoc labeled and guideline-based labeled data sets.

So apart from the explanations just mentioned, there are two other possible explanations (next to the threats to validity) for this observation. On the one hand, these discrepancies can emerge from the actual mood of the participants. Probably, in a software project setting, they may perceive single messages differently compared to this study setting. In a work situation, i.e., in a professional context, some statements might raise different feelings. On the other hand, the observation may be due to a general problem of sentiment analysis that has, to the best of our knowledge, not yet been addressed: Even though the predefined labels coincide with the median perception of a group of computer scientists, e.g., a development team, this is not necessarily true for single persons. In particular, as this kind of data base is used for the training of sentiment analysis tools, these tools also only reflect the median mood of the target group. That is, if a sentiment analysis tool is used in a team that tends towards very good or very bad mood

(i.e., that appears to be optimistic rather than pessimistic, or vice versa) it is likely that the tool does not adequately reflect the real sentiment in this particular team. Therefore, it is worth a thought to focus on calibrating sentiment analysis tools to specific teams in future research. This would help increasing the accuracy of the analyses, which would in turn increase the trust of the team in the results.

(3) Regarding the ad hoc labels of the participants, according to our results, the guideline-based data set performs better in almost all cases compared to the ad hoc labeled data set. This is somewhat surprising, as higher agreement between ad hoc labels (of the participants) and ad hoc labels (in the data set) was expected than between ad hoc and guidelines-based labels. The main difference between these two data sets is the way, the raters are asked to assign the labels to the statements. In case of the ad hoc labeling, the raters are asked to label the statements according to their perception without further thinking about it. This leads to some kind of unorganized rating process, but this way of labeling coincides with “reality”. A receiver of a message will unlikely think about how to interpret such statement (i.e., whether it is meant to be positive, negative, or neutral), but will trust his or her gut feeling. However, the guideline-based labeling process, in which the raters apply specific guidelines when assigning the polarity class to a statement, appears to be more stringent with the gut feeling of the participants in our study, as we did not tell them how to label the statements. Most of them used the tone and the content, which raise an impression by the reader, and thus reflect the gut feeling.

As Imtiaz et al. [23] have also pointed out, there seem to be different reasons for assigning a sentiment. For example, participants might assign labels based on the perception of themselves being the receiver of a message, or they might put themselves in the position of the sender. They could also label sentiments based on the content. Novielli et al. [19] did not have high Fleiss’ κ values in their emotion attribution. Their focus was on emotion recognition to build their data set. But it is debatable whether this emotion aspect is sufficient for a complete sentiment analysis, or if you still have to look at the other aspects mentioned before. Our results show that computer scientists do not seem to focus only on emotions in their perception. Or their perception of emotion differs from that of the authors. This means that tools trained with such data sets are limited in their practical applicability.

5.3. Implications

Independent of the selected labeling process, our results point to the necessity of an increased awareness of the data sets to be used when training sentiment analysis tools, as this selection strongly influences the outcome of a tool. Thus, the data base should be selected with care and should be appropriate for the pursued use case. But also after having selected an appropriate data set, it is worth a thought whether the tool still needs to be adjusted to the given context, including the team in which it should be used. This could be done, for example, by randomly examining the communication data manually and calculating the balance of negative, neutral, and positive, and matching it with that of the data set. However, this requires further research. Another option that, however, requires further research would be some kind of calibration to the team with its specific characteristics. Do we use the tool in a team that generally has a very good mood or is it a rather depressed team? Is there a lot of irony in the team communication that in addition increases the risk of misclassifications? As all these questions have been sparsely, if at all, considered in literature, further studies are required to explain how this adjustment can be achieved.

In addition, when developing sentiment analysis tools, researchers should ask themselves what they want to predict. We do not claim that the results of our study are always correct, but our findings highlight that the predefined labels of the data sets are not correct for each and every person (what the authors of the data sets do not claim). However, the accuracy of sentiment analysis tools is often tested against these predefined labels. This raises the question how meaningful such kind of validation is if the labels of the data sets do not coincide with the perceptions of the team. The gold standard data sets are a good option to test the general accuracy and performance of sentiment analysis tools, but it does not allow profound conclusions on the applicability in a specific team. And this is what researchers and practitioners should keep in mind when using such tools. The tool can always only be as good as the underlying data, but it is questionable how well the underlying data fits the team.

Summarizing, the results of our study highlight the need for awareness when applying sentiment analysis tools such that they fit the given context and the team. That is, both the origin of the training set and, hence, the application domain of the sentiment analysis tool, and the subjective labels assigned to the training data must fit the perceptions of the team. In addition, as sentiment analysis reflects subjective perceptions, results should be handled with care. The tools that produce the results are trained on subjective data (that might be made kind of objective by referring to guidelines) and, thus, the outcome should be seen as an indicator rather than the ultimate truth.

5.4. Threats to Validity

Our results are limited to our (selected) survey population, and cannot be generalized for all developers or computer scientists. In this section, we summarize the most relevant threats to validity possibly impacting our results.

Due to the location of the researchers, the vast majority of our population were non-native English speaker. Nevertheless, the statements from the data sets were entirely in English including many technical terms. To countermeasure this threat, we asked our participants about their frequency of communication in English. About two thirds of our population reported to communicate in English once a week or more, while one third communicated in English once in a while or less. Nevertheless, we assume that all participants considered their own English comprehension suitable for performing the survey. In addition, almost all technical terms from the software engineering or computer science domain are English independent of the language used.

Due to the nature of a survey study, the participants answered the survey questions at home, leading to possible distractions and interferences for individual participants.

The study was also distributed among colleagues with the request to distribute the study to other potential candidates. Therefore, it is possible that raters of the two data sets used in our paper also participated. We could not exclude this in advance, but we found it negligible due to the high number of participants.

The vast majority of over two thirds of our population were students in the computer science field, and not professional developers. We consider this threat negligible, since even gold standard data sets are often created with the annotations of computer science students (cf. [19]), and not long-standing professional developers. However, we will investigate our gathered data on group specific behavior in future research. For all that, we only included participants who identified as (prospective) computer scientists or had programming experience. We are confident that the median labels of our participants reflect the perception of an average individual in the software engineering domain.

The capability of participants to annotate a sentiment polarity to a message can be aggravated by the lack of the messages context. We only presented single, randomly selected messages from the original data sets and no course of a conversation. Some participants mentioned trying to guess the context of a response in order to be able to choose a proper sentiment polarity class for it. Although we are aware of this, we wanted to gather the initial emotional responses for the statements instead of presenting the participants with an annotation criteria beforehand and making them act accordingly when selecting the sentiment polarity classes.

6. Conclusion

Sentiment analysis tools strongly rely on pre-labeled data sets that provide polarity classes for specific statements. These polarity classes shall reflect the perception of a somewhat “typical” developer as the resulting tools are meant to be able to predict his or her perception.

In order to compare the perceptions of potential software project team members with the predefined labels, we conducted an online survey. Based on 94 data points, we compared the median perception of the participants with the predefined labels, leading to a match between the perceptions and the labels in 62.5% of the cases.

In a next step, we concentrated on single participants and evaluated their agreement with the predefined labels, leading to results ranging from systematic disagreement to almost perfect agreement. This points to a potential need to adjust the tools to the personalities of a team, which should be addressed in future research.

On average, the agreement between single participants and the group of participants is better in case of the guideline-based labeled data set compared to the ad hoc labeled one, despite the fact that the participants labeled the statements also according to their gut feeling (without any guidelines) in our survey.

These results should increase the awareness of the need to carefully select the training data set for the development of sentiment analysis tools, and to handle the results with care as perceptions are very subjective and the forecast of a sentiment analysis tool should not be over-interpreted. Summarized, the results of sentiment analysis in project contexts should be taken as a grain of salt rather than as the ultimate truth.

Acknowledgment

This research was funded by the Leibniz University Hannover as a Leibniz Young Investigator Grant (Project *ComContA*, Project Number 85430128, 2020–2022).

References

- [1] R. E. Kraut, L. A. Streeter, Coordination in software development, *Commun. ACM* 38 (3) (1995) 69–81. doi:10.1145/203330.203345.
- [2] S. Marjaie, U. Rathod, Communication in agile software projects: qualitative analysis using grounded theory in system dynamics, in: *Proc. Int'l Conf. of the System Dynamics Society* 2011, 2011.
- [3] I. R. McChesney, S. Gallagher, Communication and co-ordination practices in software engineering projects, *Information and Software Technology* 46 (7) (2004) 473–489. doi:10.1016/j.infsof.2003.10.001.
- [4] T. Niinimäki, A. Piri, C. Lassenius, M. Paasivaara, Reflecting the choice and usage of communication tools in global software development projects with media synchronicity theory, *Journal of Software: Evolution and Process* 24 (6) (2012) 677–692. doi:10.1002/smr.566.
- [5] D. Graziotin, X. Wang, P. Abrahamsson, Happy software developers solve problems better: psychological measurements in empirical software engineering, *PeerJ* 2 (2014) e289. doi:10.7717/peerj.289.
- [6] D. Graziotin, X. Wang, P. Abrahamsson, How do you feel, developer? an explanatory theory of the impact of affects on programming performance, *PeerJ Computer Science* 1 (2015) e18. doi:10.7717/peerj-cs.18.
- [7] M. Obaidi, J. Klünder, Development and application of sentiment analysis tools in software engineering: A systematic literature review, in: *International Conference on Evaluation and Assessment in Software Engineering*, ACM, Association for Computing Machinery, New York, NY, USA, 2021, p. 80–89. doi:10.1145/3463274.3463328.
- [8] M. R. Islam, M. F. Zibran, Leveraging automated sentiment analysis in software engineering, in: *2017 IEEE/ACM 14th International Conference on Mining Software Repositories (MSR)*, IEEE, 2017, pp. 203–214. doi:10.1109/MSR.2017.9.
- [9] T. Ahmed, A. Bosu, A. Iqbal, 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 (ASE)*, IEEE, Piscataway, NJ, USA, 2017, pp. 106–111. doi:10.1109/ASE.2017.8115623.
- [10] F. Calefato, F. Lanubile, F. Maiorano, N. Novielli, Sentiment polarity detection for software development, *Empirical Software Engineering* 23 (2018) 1352–1382. doi:10.1007/s10664-017-9546-9.
- [11] J. Klünder, J. Horstmann, O. Karras, Identifying the mood of a software development team by analyzing text-based communication in chats with machine learning, in: *International Conference on Human-Centred Software Engineering*, Springer International Publishing, Heidelberg, BW, DE, 2020, pp. 133–151.
- [12] M. Herrmann, J. Klünder, From textual to verbal communication: Towards applying sentiment analysis to a software project meeting, in: *2021 IEEE 29th International Requirements Engineering Conference Workshops (REW)*, IEEE, Piscataway, NJ, USA, 2021, pp. 371–376. doi:10.1109/REW53955.2021.00065.
- [13] A. Murgia, P. Tourani, B. Adams, 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, MSR 2014*, Association for Computing Machinery, New York, NY, USA, 2014, p. 262–271. doi:10.1145/2597073.2597086. URL <https://doi.org/10.1145/2597073.2597086>
- [14] M. Islam, M. Zibran, Sentistrength-se: Exploiting domain specificity for improved sentiment analysis in software engineering text, *Journal of Systems and Software* 145 (2018) 125–146. doi:10.1016/j.jss.2018.08.030.
- [15] N. Novielli, F. Calefato, D. Dongiovanni, D. Girardi, F. Lanubile, Can we use se-specific sentiment analysis tools in a cross-platform setting?, in: *Proceedings of the 17th International Conference on Mining Software Repositories*, Association for Computing Machinery, New York, NY, USA, 2020, pp. 158–168. doi:10.1145/3379597.3387446.
- [16] B. Lin, F. Zampetti, G. Bavota, M. Di Penta, M. Lanza, R. Oliveto, Sentiment analysis for software engineering: How far can we go?, in: *Proceedings of the 40th International Conference on Software Engineering, ICSE '18*, Association for Computing Machinery, New York, NY, USA, 2018, p. 94–104. doi:10.1145/3180155.3180195.
- [17] P. R. Shaver, J. C. Schwartz, D. Kirson, C. O'Connor, Emotion knowledge: further exploration of a prototype approach., *Journal of personality and social psychology* 52 (6) (1987) 1061–86.
- [18] W. G. Parrott, *Emotions in social psychology: Essential readings*, psychology press, 2001.
- [19] N. Novielli, F. Calefato, F. Lanubile, A gold standard for emotion annotation in stack overflow, in: *Proceedings of the 15th International Conference on Mining Software Repositories, MSR '18*, Association for Computing Machinery, New York, NY, USA, 2018, p. 14–17. doi:10.1145/3196398.3196453.
- [20] J. L. Fleiss, Measuring nominal scale agreement among many raters, *Psychological Bulletin* 76 (5) (1971) 378–382. doi:10.1037/h0031619.
- [21] J. Cohen, A coefficient of agreement for nominal scales, *Educational and Psychological Measurement* 20 (1) (1960) 37–46. doi:10.1177/001316446002000104.
- [22] T. Zhang, B. Xu, F. Thung, S. A. Haryono, D. Lo, L. Jiang, Sentiment analysis for software engineering: How far can pre-trained transformer models go?, in: *2020 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, IEEE, Adelaide, SA, Australia, 2020, pp. 70–80. doi:10.1109/ICSME46990.2020.00017.
- [23] N. Imtiaz, J. Middleton, P. Girouard, E. Murphy-Hill, Sentiment and politeness analysis tools on developer discussions are unreliable, but so are people, in: *Proceedings of the Third International Workshop on Emotion Awareness in Software Engineering*, Association for Computing Machinery, New York, NY, USA, 2018, p. 55–61. doi:10.1145/3194932.3194938.
- [24] J. Ding, H. Sun, X. Wang, X. Liu, Entity-level sentiment analysis of issue comments, in: *Proceedings of the 3rd International Workshop on Emotion Awareness in Software Engineering, SEmotion '18*, Association for Computing Machinery, New York, NY, USA, 2018, p. 7–13. doi:10.1145/3194932.3194935.
- [25] G. Uddin, F. Khomh, Automatic mining of opinions expressed about apis in stack overflow, *IEEE Transactions on Software Engineering* 47 (3) (2021) 522–559. doi:10.1109/TSE.2019.2900245.
- [26] S. Mohammad, A practical guide to sentiment annotation: Challenges and solutions, in: *Proceedings of the 7th workshop on computational approaches to subjectivity, sentiment and social media analysis*, Association for Computational Linguistics, 2016, pp. 174–179. doi:10.18653/v1/W16-0429.

- [27] C. Robson, K. McCartan, Real World Research, John Wiley & Sons, Inc., New York, NY, USA, 2015.
- [28] J. Wu, C. Ye, H. Zhou, Bert for sentiment classification in software engineering, in: 2021 International Conference on Service Science (ICSS), 2021, pp. 115–121. doi:10.1109/ICSS53362.2021.00026.
- [29] Z. Chen, Y. Cao, H. Yao, X. Lu, X. Peng, H. Mei, X. Liu, Emoji-powered sentiment and emotion detection from software developers' communication data, ACM Trans. Softw. Eng. Methodol. 30 (2). doi:10.1145/3424308.
URL <https://doi.org/10.1145/3424308>
- [30] M. Herrmann, M. Obaidi, L. Chazette, J. Klünder, Dataset: On the Subjectivity of Emotions in Software Projects: How Reliable are Pre-Labeled Data Sets for Sentiment Analysis? (Jun. 2022). doi:10.5281/zenodo.6611728.
URL <https://doi.org/10.5281/zenodo.6611728>
- [31] Wes McKinney, Data structures for statistical computing in python, in: Proceedings of the 9th Python in Science Conference, SciPy, Austin, TX, USA, 2010, pp. 56–61. doi:10.25080/Majora-92bf1922-00a.
- [32] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, T. E. Oliphant, Array programming with NumPy, Nature 585 (7825) (2020) 357–362. doi:10.1038/s41586-020-2649-2.
- [33] J. D. Hunter, Matplotlib: A 2d graphics environment, Computing in science & engineering 9 (3) (2007) 90–95. doi:10.1109/MCSE.2007.55.
- [34] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.
- [35] S. S. Shapiro, M. B. Wilk, An analysis of variance test for normality (complete samples), Biometrika 52 (3–4) (1965) 591–611. doi:10.1093/biomet/52.3-4.591.
- [36] Student, The probable error of a mean, Biometrika 6 (1) (1908) 1–25. doi:10.2307/2331554.
- [37] D. Rey, M. Neuhäuser, Wilcoxon-Signed-Rank Test, Springer, Berlin, Heidelberg, 2011, pp. 1658–1659. doi:10.1007/978-3-642-04898-2_616.
- [38] W. Haynes, Bonferroni Correction, Springer, New York, NY, USA, 2013, pp. 154–154. doi:10.1007/978-1-4419-9863-7_1213.
- [39] J. Landis, G. Koch, The measurement of observer agreement for categorical data., Biometrics 33 1 (1977) 159–74. doi:10.2307/2529310.
- [40] Nicole Novielli, Daniela Girardi, Filippo Lanubile, A benchmark study on sentiment analysis for software engineering research, in: 2018 IEEE/ACM 15th International Conference on Mining Software Repositories (MSR), MSR '18, Association for Computing Machinery, 2018, pp. 364–375. doi:10.1145/3196398.3196403.