e-Informatica Software Engineering Journal, Volume 16, Issue 1, 2022, pages: 47–68, DOI 10.37190/e-Inf220102

Analysis of Factors Influencing Developers' Sentiments in Commit Logs: Insights from Applying Sentiment Analysis

Rajdeep Kaur*, Kuljit Kaur Chahal*, Munish Saini**

*Department of Computer Science, Guru Nanak Dev University, Amritsar, India

**Department of Computer Engineering and Technology, Guru Nanak Dev University, Amritsar, India

rajdeep.rsh@gndu.ac.in, kuljitchahal.cse@gndu.ac.in, munish.cet@gndu.ac.in

Abstract

Background: In the open source software paradigm, software development depends upon efforts of volunteer members that are geographically dispersed and collaborate with each other over the Internet. Communication artifacts like mailing lists, forums, and issue tracking systems are used by developers for communication. The way they express themselves through these communication channels greatly influences their productivity, efficiency of development activities, and survival of the project as well. Therefore, it is essential to understand affective state of developers' contributions to make software engineering more effective.

Aim: This study examined commit logs of seven GitHub projects to analyze developers' sentiments. This study also investigated the relationship of developers' sentiments in commit logs with team size of project, type of change activity, and contribution volume.

Method: Sentiments of developers are calculated using SentiStrength-SE tool that is specialized in software engineering domain.

Results: Our findings revealed that the majority of sentiments conveyed by developers in commit logs were neutral. Furthermore, we found that team size, change activity, and commit contribution volume influenced sentiments conveyed in commit logs.

Conclusion: Our findings will help project managers to better understand developer sentiments while performing different software development tasks/activities. It will be beneficial in improving developer productivity and retention.

Keywords: human factors in software development teams, software developer, developers' sentiment, sentiment analysis, commit logs, developer activity type, and team size

1. Introduction

Sentiments of software developers greatly influence the quality and productivity of developed software [1]. Prior studies confirm that emotions impact task quality, productivity, creativity, group rapport, and job satisfaction [2]. Due to advancements in Natural Language Processing (NLP) and significance of human computer interaction, research associated with sentiments and emotional aspects of software developers' communication is gaining more traction in the software engineering domain.

Submitted: 16 Jun. 2021; Revised: 10 Sep. 2021; Accepted: 23 Sep. 2021; Available online: 15 Dec. 2021

Sentiment analysis is an opinion mining method used to identify people's sentiments, views, evaluations, feelings, attitudes, and appraisals about products, organizations, services, topics, events, issues, individuals, and their attributes [3]. It is basically used to classify opinion in written text into positive, negative, and neutral. Sentiment analysis was first introduced by Liu et al. [3]. Originally, sentiment analysis was used to detect the polarity of small text posted in product reviews, movie reviews, tweets, and microblogs [4]. In recent times, this technique is widely adopted by software engineering community and applied to various software artifacts like commit logs [4–7], mailing lists messages [8], issue comments [9, 10], code reviews [11], bug reports [12]. In order to better support developers during software development activities and understanding the social factors that affect productivity and retention, it is necessary to understand their sentiment in various software development tasks. This information may help managers of OSS software projects to better support developers with tools during software development and resolve the issues related to various tasks. Thus, it will help in improving developers' productivity as well as retention.

In the present field of study, we observed significant work done by different researchers to examine developers' sentiments in commit messages of OSS (Open Source Software) [4–7], etc. However to the best of our knowledge, none of them analyzed the relationship of type of change activity performed by developers, their commit contribution, and team size of a project (Large, Medium, and Small) with sentiments expressed by developers in commit logs. Our work also looks into the evolution of sentiments with respect to time. Thus lack of research in the domain motivated us to conduct this research work.

In this work, we investigated the sentiments of developers conveyed in commit logs. Sinha et al. [5] also examined the developers' sentiments in commit logs and relate the sentiments in commit messages with the day of week and number of changed files but our study has a different objective. We studied the developers' sentiments across seven well-known GitHub projects to examine the impact of project team size on developers' sentiments. Furthermore, type of change activity executed by developers was considered and then analyzed the impact of Type-1 (add + modify), Type-2 (delete + modify), and Type-3 (add + delete + modify) activity on the sentiment of developers projected in the commit logs. The existing literature reported three types of change activity viz. addition, deletion, and modification [4]. We grouped the individual change activity into combinations of two or three file change activities to create our own classification scheme. Apart from this, the authors also investigated the association between commit contribution and volume of sentiment. Sentiment volume is percentage of sentiments (positive, negative, and neutral) conveyed by individual developer in the commit log and commit contribution size is percentage of commits made by individual developer. Besides, our work also examined the evolution of sentiment in the project with respect to commits that is not taken into account by Sinha et al. [5]. To achieve the aforementioned objective, we formulated the following research questions:

RQ1: What are the overall developers' sentiments in the commit logs?

– Developers' inactivity in the project is associated with their negative and positive mood value [12]. Thus an understanding of developers sentiment attached to commit activity might be helpful for project managers in introducing measures to manage developers' sentiments that may ensure the stability of developers.

RQ2: Is there any relation between sentiments and team size of a project?

– The accomplishment of the large project relies on a large number of developers and a long development period. Developers working with a large code base may lead to negative emotions in the project due to workload and stress in managing a large code base. Moreover, staffing and task allocation is a complex task in large projects. Thus,

this makes it difficult to manage projects, and the decision of managers largely influences the mood of developers. Thus an understanding of impact of project team size can be used to effectively manage developers' emotions in the project that may lead to high productivity and improved job satisfaction [13].

RQ3: Does the type of change activity performed by a developer impact their sentiments in commit messages?

– Developers who convey positive emotions while executing a particular development task might be more efficient and fast in accomplishing a task [14] that will reduce cost of software. Thus understanding developers' sentiments attached to a particular task can be helpful in effective task allocation. For example, making tasks (read issues in an issue tracking system) simple to understand, and easy to solve by decomposing complex issues into smaller ones can improve developer productivity, and sentiment in commit logs.

RQ4: Is there any relation between developer sentiment volumes and commit contribution size?

– Understanding emotional state of developers involved in high or low commit activity may help project managers to effectively distribute workload among developers and increasing development activity as well as boosting neutral or positive sentiments.

RQ5: How has sentiment in the commit logs evolved over the period of time?

– Analyzing the evolution of sentiments, we can identify trends in sentiment expression in commit logs. Is it getting negative or positive? A particular time slot when sentiments in commit logs are shifting direction e.g. becoming more positive, we can identify the reasons and try to maintain that state. For example, it has been observed in this study that reduced negativity in commit logs coincides with launch of the Github platform in 2008. Managers can take motivational steps to boost developers that may increase their retention in the project.

RQ1 aims to identify general developers' sentiments conveyed in commit messages. RQ2 aims to discover the impact of team size on the sentiments expressed by developers in commit messages. RQ3 identifies the association between three types of changes activities (Type-1, Type-2, Type-3) performed by developers and their expressed sentiments. Type-1, Type-2, and Type-3 are combinations of two or more individual file change types (addition, deletion, and modification). RQ4 intends to ascertain the impact of developers commit contribution on sentiment volume. Sentiment volume is defined as a percentage of Positive, Negative, and Neutral sentiments conveyed by each developer in the commit log, and commit contribution is a percentage of commits made by each contributor in the project. RQ5 examined the evolution of sentiments with respect to the number of commits made by developers over the period of time.

Our study uses the Sentistrength-SE [15] tool to detect polarity of sentiments conveyed in commit logs messages. This tool used lexical approach and domain dictionary and specially designed for software engineering text.

The remainder of the paper is organized as following: Existing work related to current study is discussed in Section 2. Description of data collection methodology along with detail of sentiment analysis approach used to detect sentiments of developers in commit logs is presented in Section 3. The results of study are discussed in Section 4. Discussion is presented in Section 5. Some Threats to Validity are described in Section 6. Conclusions along with some future directions are presented in Section 7.

2. Related literature

Many studies have been conducted by researchers and practitioners in the past to analyze the developers' sentiments in OSS code repositories and related artifacts. They examined developers' sentiments in different software artifacts such as commit logs, commit comments, mailing list messages, and GitHub security debates. A summary of the related literature is presented in Table 1.

Some researchers evaluated the performance of SE-specific sentiment analysis tools, compared them in terms of accuracy, and proposed techniques to improve existing sentiment analysis tools. Novielli et al. [16] in 2021 presented a replication study to evaluate the performance of SE-specific tools. Sun et al. [17] proposed sentence structure to improve sentiment analysis in software engineering text. Biswas et al. [18] in 2020, investigated the effectiveness of a customized language representation model known as BERT and Novielli et al. [19] assessed the performance of four SE domain specific tools viz. Senti4SD, SentiCR, SentiStrength-SE, and DEVA in cross-platform. M. R. Wrobel [20] investigated the influence of adoption of lexicons on emotion mining in SE artifacts.

In the year 2021, Martin Obaidi and Jil Klünder [21] presented a systematic literature review of sentiment analysis tools designed for and applied in a software engineering context. This study explored sentiments analysis tools used in the software engineering field, utilized data sets, application areas of sentiment analysis tools, and problems faced at the time of developing such kinds of tools.

Some researchers explored the sentiment variation based on different factors and also examined the association of sentiments with various factors. Huq et al. [22] in 2020 examined the relation of sentiments with software bugs. In the same year, Kaur and Chahal presented investigation of developers' sentiments in commit comments [23]. In the year 2019, Paul et al. [11] analyzed the code review data of five open source projects to investigate the difference in expression of sentiments based on the gender of developers during various software development tasks. In the year 2018, Bharti and Singh [24] surveyed 20 software professionals to examine the developers' sentiments associated with code cloning practices. Islam and Zibran [7] studied the variance in emotion in commit messages that are related to bug introduction and bug fixing activities. Singh et al. [5] have analyzed the 3,171 commit messages that are related to refactoring activities to investigate the impact of 15 different code refactoring tasks on developers' sentiments. This study identified that the developers' sentiments are more negative during refactoring activities. Souza and Silva [25] examined the relationship between sentiments of developers and build breakage in a Travis CI (continuous integration). Sinha et al. [5] investigated the developer sentiment in the commit logs of GitHub projects and studied the association among developer sentiment and day of the week. They also examined the correlation between developer sentiment and the number of files changes performed by the developer in the commits. This study demonstrates that most of the sentiments projected by developers in the commit log were neutral. The negative sentiments are 10% higher than the positive and the majority of the negative sentiment was detected on Tuesday.

Islam and Zibran [13] investigated sentiments variation based on different types of tasks executed by developers, development period, in different size projects, and impact of emotions on software artifacts (i.e., length of commit message). Garcia et al. [12] analyzed the data of the bug tracking system and mailing list to examine the association between emotions and contributor activity.

Guzman et al. [4] examined commit comments of GitHub projects to investigate the relation of developer sentiment with the programming language used by the project, time

Table 1. Summary of related studies

Author and year	Scenario of motivation	Possible extension
Huq et al. (2020)	Examined the correlation between sentiments and software bugs	The relationship between sentiments and three types of file change activity can be explored.
Paul et al. (2019)	Examined the sentiments of developers in code review comments.	Developers' sentiments can be explored in commit logs messages.
Sinha et al. (2018)	Investigated the relation between the number of file changes and developers sentiments.	Relation between different combinations of file change can be explored.
Singh et al. (2017)	Examined the impact of software code refactoring activities on the sentiments of developers.	The impact of commit contribution on developers' sentiments can be explored.
Tourani et al. (2014)	Explore the existence of positive and negative emotions in user and developer mailing lists.	Commit logs can be explored to detect developer sentiments and various factors influencing sentiments.
Guzman et al. (2014)	Explored the association of emotion with team geographical location and day and time of the week.	Relation of sentiments with team size can be explored. The evolution of sentiments with respect to the number of commits over time can be explored.
Garcia et al. (2013)	Ascertain the association between emotions and contributor activity.	The relation of commit contribution with developers' sentiments can be explored.
Md Rakibul Islam and Minhaz F. Zibran, (2016)	Examined the impact of project and team size and length of commit message on emotional states of developers.	The impact of large, medium, and small team size projects on the sentiments of developers can be explored.
Pletea et al. (2014)	Explored the emotional expression in security discussions by analyzing commits and pull request comments.	Commit logs messages can be analyzed to explore sentiment expressed in different combinations of change activities.
Khan et al. (2010)	Analyzed the effect of emotions on software developers' debugging performance.	The impact of sentiments on commit contribution can be investigated.
Muller and Fritz, (2015)	Investigated developers' emotions and progress on change tasks by conducting lab study.	Emotions conveyed in software artifacts such as commit log can be explored.
Graziotin et al. (2014)	Explored the connection between developer emotion and their ability to solve analytical problems.	The association of sentiments with different file change activities can be explored.
Michal R. Wrobel, (2013)	Conducted a survey to investigate developers' emotions in the software development process and impact of emotions on performance.	Software artifacts such as commit logs can be examined to investigate developers' conveyed emotions.

and day of the week when the comment was written, team dispersal, and project approbation. This work revealed that java projects have more negative comments. The more positive comments are detected in projects having distributed teams and Monday was the most negative day for sentiments.

Tourani et al. [8] presented a study to investigate the presence of positive and negative emotions in user and developer mailing lists. This study found that both types of mailing lists have positive as well as negative sentiments and have a different focus.

Pletea et al. [26] examined sentiments associated with security discussions in commits and pull requests. This study identified that negative emotions are higher in security debates in comparison to non-security discussions. Khan et al. [27] have analyzed the impact of emotions on the debugging performance of software programmers. Müller and Fritz [28] presented a study on developers' emotions and progress on change tasks. Graziotin et al. [29] examined the association between developer emotion and their ability to solve analytical problems. They found that happy software developers are better at solving analytical problems. In the year 2013, Wrobel [30] presented a study on developers' emotions in the software development process by conducting a survey.

To the best of our knowledge, the work presented in the past does not explore the impact of team size of the project, type of change activity, and commit contribution on sentiments of developers. The work presented in this paper is motivated by Sinha et al. [5]. This study investigated the relation of the day of week and number of changed files with developers' sentiments. But this study does not explore the association of combinations of change activity type and commit contribution with developers' sentiments. One another study presented by Guzman et al. [4] examined the sentiments expressed by developers in commit comments and investigate their association with different factors like time and weekday, project approval, coding language, and team geographical distribution. But this study does not consider the team size of the project and its association with developers' sentiments [4]. Thus lack of research in the field motivated us to conduct this research work. Our work examined the whole commit logs of seven GitHub projects to analyze sentiments of software developers projected in commit logs and investigate the effect of team size, type of change activity, and commit contribution on the developers' sentiments. Furthermore, we also look into the evolution of sentiments to identify how these changes across the years along with the number of commits. We utilized the SentiStrength-SE tool to perform sentiment analysis. We selected this tool because it is the first domain-specific tool specially designed to detect sentiments in a software engineering context and provides better accuracy in comparison to the existing domain-independent sentiment analysis tools/toolkits [31].

3. Analysis methodology

In this section, we provide a description of the dataset along with details of the approach used to conduct sentiment analysis.

3.1. Data collection

GitHub is a popular version control and project management system that provides multiple collaborative artifacts viz commits, issues, and pull requests to contributors [32]. We extracted the data of seven GitHub projects. The projects were selected based on popularity, size, number of commits, number of contributors involved, long project history (more than 10 years), and having a valid Git (distributed version control system) repository. The projects have creation dates from 1972 to 2007. Table 2 describes the quantitative details of the projects. An overview of the selected projects is given below.

Table 2. Detail description of projects

Sr. No.	Name	Project size (in lines of code)	Number of stars	Number of commits	Number of developers	Start date	End date
1.	PostgreSQL	1,113,634	8,406	66329	51	Jul. 1996	Feb. 2019
2.	Glibc	1,305,634	547	49216	538	Jan. 1992	Feb. 2019
3.	Eclipse-CDT	1,498,813	141	30651	260	Jun. 2002	Feb. 2019
4.	GNUCash	2,361,864	1923	25372	185	Nov. 1997	Feb. 2019
5.	WordPress	1,549,456	15,135	44388	96	Apr. 2003	Feb. 2019
6.	Firebug	492,078	1,289	13060	47	Aug. 2007	Oct. 2017
7.	Rhino	806,709	2,896	3903	82	Apr. 1999	Feb. 2019

PostgreSQL is an open source RDBMS (relational database management system). Glibc is a GNU C library most commonly used by GNU/Linux system. Eclipse-CDT is an IDE (integrated development environment) for developing programs in C and C++. GNUCash is accounting software developed for individual and small businesses. WordPress is a PHP and MySQL based content management software. Firebug is a web browser extension for Mozilla Firefox. Rhino is an open source JavaScript implementation that is completely written in Java. Generally, scripting for end users is implemented in java application. We accessed the repositories of the projects from GitHub[1] or git[2]. The Git Bash tool was utilized to clone project repositories to the local machine. The commit logs of the projects were retrieved using the git log command. Commit logs of all selected projects were analyzed from their beginning to till February 2019. In case of Firebug ending period is October 2017.

3.2. Sentiment analysis

There are a variety of sentiment analysis tools viz. SentiStrength [33], StafordNLP [34], and NLTK [35], while most of them do not focus on technical text. As these tools are designed for non-technical text such as movie reviews or blogs posted on social networking sites such as twitter, their results are erroneous for technical artifacts in the Software Engineering (SE) domain [36]. Therefore, domain-specific techniques provide better accuracy to detect sentiments in software engineering text.

We used sentiment analysis tool SentiStrength-SE proposed by Islam and Zibran [15] to perform sentiment analysis on commit logs. Similar choice is made by Md Rakibul Islam and Minhaz F. Zibran in Software engineering domain to extract emotional score from commit messages [7]. Using SentiStrength as the baseline, this tool implements a lexical based approach and domain specific dictionary. We selected this tool because it is a first SE specific tool specially designed for Software Engineering to conduct sentiment analysis and it outperforms the existing domain-independent tools/toolkits [31]. SentiStrength-SE tokenizes the text into words and assigns a score to each word that conveys the underlying sentiment. The words with positive sentiment receive a score between $+1$ to $+5$ and words with negative score range between -1 to -5. The neutral score of words ranges between $+1$ to -1. The scoring is generated using a sentiment dictionary that includes the predetermined polarity score of sentiment words and phrases [32]. Sentistrength-SE

[1]https://github.com
[2]https://git-scm.com

provides maximum positive and maximum negative score of each sentence. The final score of sentence is calculated by adding maximum positive and maximum negative score by following the approach used by jongjelling et al. [37]. The methodology used for sentiment analysis is illustrated in Figure 1.

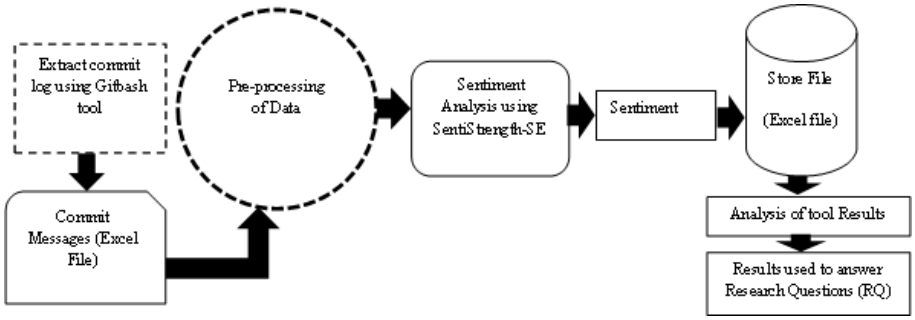


Figure 1. Methodology used for sentiment analysis

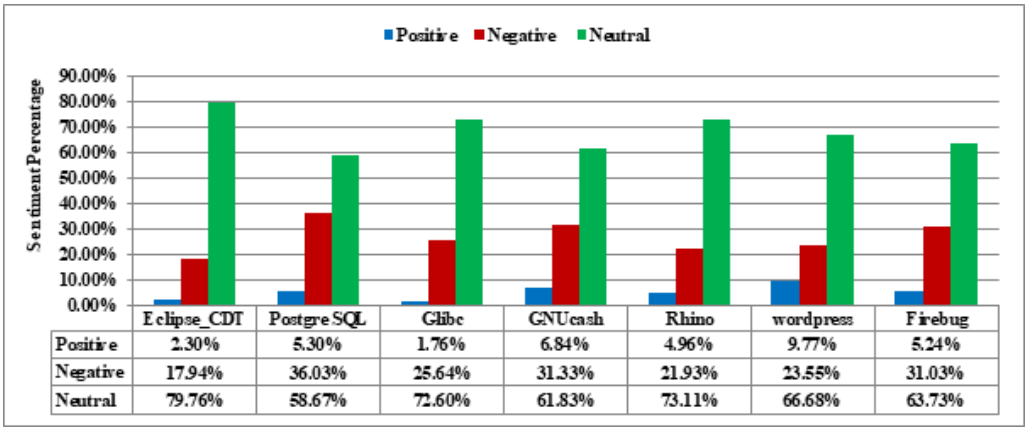
Firstly, we extracted the commit log using the git log command available in the Git Bash tool and saved the commit log data in CSV format. In the next step, extracted the commit messages and pre-processed the collected data to remove stop words, white spaces, non-alphanumeric symbols/characters, and punctuation marks from the text. In addition, also removed code, URLs, and system generated messages, e.g., error messages. Then sentiment analysis is performed using SentiStrength-SE tool. Finally, we get the sentiment score of each commit message.

4. Results and analysis

In this section, we report the results of each research question formulated in Section 1.

RQ1: What is the general developer sentiment in the commit logs?

We examined a total of 86,515 commit messages of seven OSS projects to analyze developers' sentiments in commit logs. Commit logs of all selected projects were analyzed from their beginning until the last observation date set by this study (Refer Table 2). Results of sentiment analysis using SentiStrength-SE are illustrated in Figure 2. Table 3



	Eclipse_CDT	Postgre SQL	Glibc	GNUcash	Rhino	wordpress	Firebug
Positive	2.30%	5.30%	1.76%	6.84%	4.96%	9.77%	5.24%
Negative	17.94%	36.03%	25.64%	31.33%	21.93%	23.55%	31.03%
Neutral	79.76%	58.67%	72.60%	61.83%	73.11%	66.68%	63.73%

Figure 2. Sentiments across all projects

Table 3. Commit messages with positive, negative, and neutral sentiment

Sentiment	Commit message	Final sentiment score
Positive	Add test case for pthread_sg etname_np	1
	some fixes to project description manager and build system to allow EFS hosted projects to function better	2
	Generic implementation of red-black binary tree It's planned to use in several places	2
Negative	Oops did inadvertent branch	–2
	Bugzilla 218654 This commit shows some files contain errors This is because they are being compiled against M4 I will rebuild against M5 shortly I did diff of the files and changes are exactly what I wanted They will compile against HEAD and M5 when that is resolved	–1
	Oops Removing unneeded System.err.println foo	–1
Neutral	Build/TestToolsMove WP_UnitTestCase_BaseassertPostConditions to more appropriate place	0
	New ScannerInfoProvider extension point allowing providers to be associated with build commands in the project description	0

presents some examples of positive, negative, and neutral commits from GitHub dataset. As noted in Figure 2, all projects (Eclipse-CDT, PostgreSQL, Glibc, GNUCash, Rhino, Firebug, and WordPress) have a higher proportion of the neutral sentiment as compared to the negative and positive ones. Eclipse-CDT has the highest neutral (79.77%) sentiments, and lowest negative sentiments in comparison to other projects. PostgreSQL logs have the most negative (36.03%) sentiments. The proportion of positive sentiment is the lowest in all projects as compared to neutral and negative sentiments. WordPress logs have the highest positive (9.77%) sentiments and Glibc has least positive sentiments (1.76%). Our findings clearly indicate that the overall sentiments expressed in commit logs were neutral.

Our findings clearly indicate that majority of commits in commit logs are neutral in comparison to negative and positive. There is lowest percentage of positive commits than negative and positive ones. The main reason for high neutrality in the commits may be that commits are different from online reviews and tweets. However, a small amount of commit messages in commit logs have different types of affective states than review comments posted online. People express their satisfaction and dissatisfaction about a product by writing reviews whereas software developers write commit messages when they submit their work output in the form of code in a repository. The submission may include some code and URLs while writing commit messages without mentioning any affective involvement that makes the sentiments conveyed in commits more neutral. Moreover, commit messages include many technical terms that do not have any sentiment manifestation. Therefore, it could be another reason for the neutral sentiments in commit logs. Moreover, commit messages include many technical terms that do not have any sentiment manifestation. Therefore, it could be another reason for the neutral sentiments in commit logs.

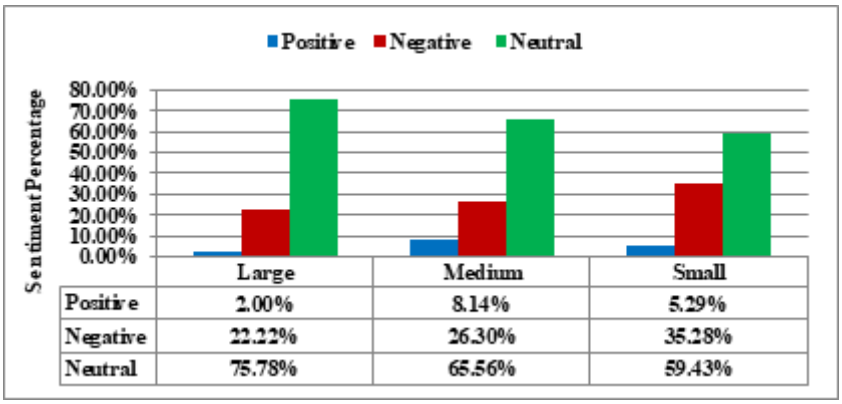
RQ2: Is there any relation between sentiments and team size of a project?

In this research question, our objective is to ascertain if size of the team in a project has any impact on sentiments expressed by developers in commit logs. We categorize the projects into large, medium, and small based on the number of contributors involved in each project (see Table 4) as recommended by Becher et al. [38]. We consider participant as developer who made at least one commit in the project. The projects having 40 to

Table 4. Project size boundaries

Parameters	Minimum developers	Maximum developers	Project Name
Small	40	60	PostgreSQL, Firebug
Medium	61	200	GNUCash, WordPress, Rhino
Large	201	∞	Glibc, Eclipse-CDT

60 developers are classified as small, projects with 61 to 200 developers as medium, and projects comprising more than 201 developers as large projects (see Table 4). Becher et al. present a study to analyze number of contributors in a random sample of projects included in the GNU/Linux distribution [38]. We followed the partition proposed by Becher et al. [38] to construct project size boundaries that are presented in Table 4 and categorize the projects into small, medium, and large based on these size boundaries [38]. The sentiment score of large, medium, and small projects is presented in Figure 3.



	Large	Medium	Small
Positive	2.00%	8.14%	5.29%
Negative	22.22%	26.30%	35.28%
Neutral	75.78%	65.56%	59.43%

Figure 3. Sentiment in projects with Large, Medium, and Small number of contributor

As shown in Figure 3, all three categories (Large, Medium, and Small) of projects have high count for neutral sentiments than negative and positive sentiments. The Projects with a large number of contributors have more neutral sentiments (75.78%) as compared to projects having medium and small number of contributors. But, we see an opposite trend in projects having medium and small number of contributors. In projects with medium team size, the percentage of positive sentiments (8.14%) expressed is higher and the percentage of negative sentiments is lower than projects with small team size. Lastly, negative sentiment is maximum in projects with small team size.

Due to the fluctuating number of team members in an OSS project over a period of time, it is worthy to relate sentiments in commit logs with the number of active developers in a smaller unit of time. For this, we identified active developers in the projects in each year of their lifetimes. To identify active developers, observation period is chosen for each project is January 2018 whereas for Firebug it is September 2016. Developers those show any activity after January 2018 is considered active. In case of Firebug developers having any activity after September 2016 considered active. Sentiments are mapped to number of active developers in each year to determine the relation of sentiment with active developers (team size). Results are presented in Figure 4. As shown in Figure 4, findings of Eclipse-CDT, PostgreSQL, Rhino and firebug indicate that neutral sentiments are high with large or small team. In case of negative sentiments large team indicate low negativity whereas small team indicate high negativity in sentiments. Results of Eclipse-CDT, PostgreSQL, Glibc,

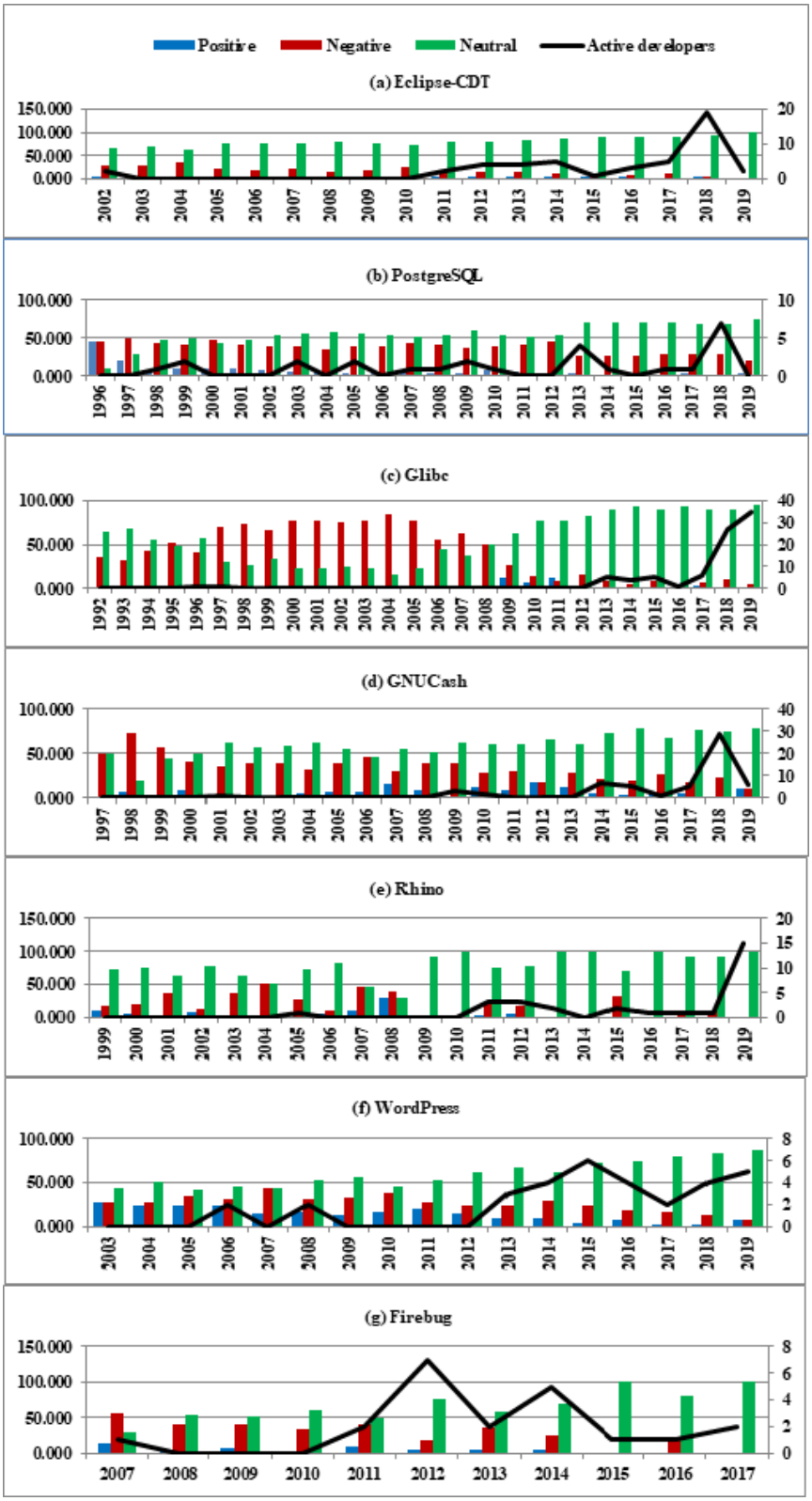


Figure 4. Sentiments and number of active contributors
(a) Eclipse-CDT, (b) PostgreSQL, (c) Glibc, (d) GNUCash, (e) Rhino, (f) WordPress, and (g) Firebug

WordPress, and Firebug indicate that positive sentiments are low with large team and high with small team. When we compared these results with Figure 3, we observed similar trend.

To confirm our results, we applied Pearson Correlation between number of active developers in each year and sentiments. The results of Pearson Correlation are presented in Table 5. In case of Glibc, Eclipse-CDT, GNUCash, and WordPress, we found strong positive correlation (>0.47) between neutral sentiments and number of active developers. In Glibc, Eclipse-CDT, and WordPress, we found strong negative correlation between number of active developers and negative sentiments. No significant correlation is found between number of active developers and positive sentiments. Only Wordpress shows significant correlation between number of active developers and positive sentiments.

Table 5. Pearson co-relation between active developers and sentiment
(* means correlation is significant at the 0.05;
** means correlation is significant at 0.01 level)

Project	Positive	Negative	Neutral
Glibc	–.117	–.446*	.482**
Eclipse-CDT	.036	–.533*	.498*
GNUCash	–.218	–.366*	.474*
WordPress	–.737**	–.634**	.769**
Rhino	–.186	–.296	.309
PostgreSQL	–.273	–.340	.349*
Firebug	.010	–.278	.228

Our findings clearly indicate that projects with large team size have more neutral sentiments. One main reason for high neutrality in the sentiments may be that developers in a large team are more formal and used many technical terms while writing commits that do not have any affective state. Moreover large teams may have laid down some formal coding guidelines. Therefore, it makes the sentiments more neutral. In small team size setup, projects have more negative sentiments in commit logs. It may be developers are less formal in a small team, or it could also be due to work pressure. There is need to look at it in the future work.

Our finding confirms that projects with different team size show different trends in the sentiments. Hence, team size of a project influences the sentiments expressed by developers in its commit logs.

RQ3: Does the type of change activity performed by a developer impact their sentiments in commit messages?

In this research question, we intended to recognize the relation between type of change activity performed by developers and sentiments expressed by them in the commit messages. There are three types of code change activities, i.e., addition, deletion, and modification [4], which can be combined in various ways to change a program. For example, some change may require adding new code along with modification of the existing lines of code. Based on these three types of file changes, we create our own classification by making the following combinations of file change types: add + modify, delete + modify, add + delete + modify. The motivation for these combinations is the evidence in the Software Engineering literature that modification of existing code is more difficult than adding new or deleting existing code. Creating new code is fun, but changing the existing one is hard.

We select five projects (PostgreSQL, Eclipse-CDT, Firebug, GNUCash, and WordPress) out of seven projects based on three types of activities performed by developers. For our analysis, we classify developers according to three types of change activities such as Type-1 (add + modify), Type-2 (delete + modify), and Type-3 (add + delete + modify) and analyze developers' sentiments based on the type of change performed by them.

The results of sentiment analysis based on three types of change activities are presented in the Figure 5. From these results, we observed that neutral sentiments have minimum occurrences for Type-3 activity. Also this is the activity which involves the most negative sentiments. Type-2 activity indicates high neutral sentiments (see results of PostgreSQL,

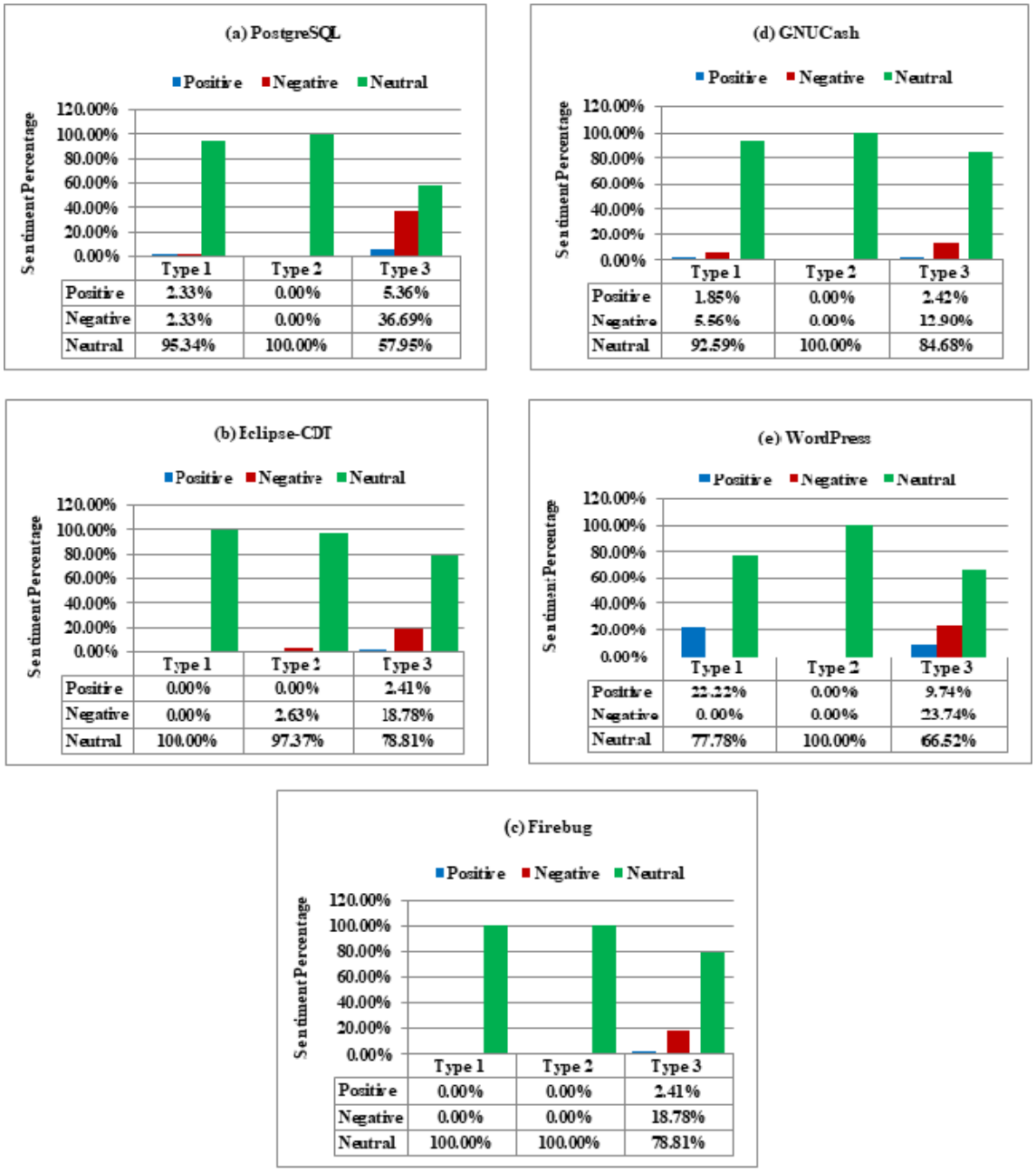


Figure 5. Sentiment and type of change activity:
(a) PostgreSQL, (b) Eclipse-CDT, (c) Firebug, (d) GNUCash, and (e) Wordpres

GNUCash, and WordPress indicated in Figure 5) in comparison to Type-1 and Type-3 activity.

To conclude, RQ3 results, the sentiments conveyed by developers in commit messages are influenced by the type of change activity performed by them. It shows that more negative and less neutral expression is put with Type-3 (add + delete + modify) activity. The reason for this could be that in Type-1 and Type-2 activities developers perform two operations in each while in Type-3 activity they perform 3 operations that means more complex work and it may make the sentiments more negative in comparison to Type-1 and Type-2. From these results, we inferred that when developers are involved in more than two activities, they express more negative expressions in the commit messages.

RQ4: Is there any relation between developer sentiment volume and commit contribution?

In this research question, we want to determine the association between sentiment volume and commit contribution. In order to achieve this goal, we analyzed developers' sentiments in commit logs and calculate the commit contribution of the top ten contributors. Commit contribution is the percentage of commits made by each individual contributor in a project. We calculate the commit contribution by dividing the total commits of each individual contributor by total number of commits made in the project. Sentiment volume, formulized in the same way as commit contribution size, is percentage of sentiments (Positive, Negative, and Neutral) conveyed by each individual developer in the commit log. We also compute the sentiment volume of each contributor by dividing individual contributor total sentiment (Positive, Negative, and Neutral) by total sentiments of the project. The formulas used for calculation of commit contribution and sentiment volume are as mentioned below:

$$Commit\ Contribution = \frac{Total\ Commits\ of\ Individual\ Contributors}{Total\ Number\ of\ Commits\ in\ the\ Project}$$

$$Sentiment\ Volume = \frac{Contributor\ Sentiment}{Total\ Sentimentin\ the\ Project}$$

$$Positive\ Sentiment\ Volume = \frac{Contributor\ Total\ Positive\ Sentiment}{Total\ Positive\ Sentiments\ of\ Project}$$

$$Negative\ Sentiment\ Volume = \frac{Contributor\ Total\ Negative\ Sentiment}{Total\ Negative\ Sentiments\ of\ Project}$$

$$Neutral\ Sentiment\ Volume = \frac{Contributor\ Total\ Neutral\ Sentiment}{Total\ Neutral\ Sentiments\ of\ Project}$$

We map sentiment volume to the commit contribution. The sentiment volume (positive, negative, and neutral) of top ten contributors along with their commit contribution is presented in Figure 6. Negative sentiment can be attributed to lead contributors in almost every project. It may be due to the project deadlines or other challenges such as quality evaluation that core contributors evoke more negative sentiment. It could also be due to the status in the team or control over the project that invites negative sentiments. For some projects such as PostgreSQL, and Firebug, neutral sentiment is clearly higher for developers with least contributions. But there is no such clear pattern for other sentiments and for

other projects. Rather, the sentiment is negative irrespective of the contribution size. So there is no trend that indicates any relation between contribution size and sentiment volume except that large contributors elicit more negative sentiment.

To conclude RQ4 results, we observed that the commit contribution of the developer influences their sentiments in the commit log. We noticed different trends in sentiments with respect to commit contribution. In most of the projects, the developers have more negativity in sentiment when their contribution is large and contributors with small commit contributions have a more positive sentiment. This implies that high commit activity causes negative sentiments in the project.

We applied Pearson Correlation to identify the correlation between commit contribution and sentiments (positive, negative, and neutral). In WordPress and firebug, we found a strong positive correlation (Pearson's correlation test above 0.70) between commit contributions and the positive sentiments whereas Eclipse-CDT has a strong negative correlation ($>$–0.70). The GNUcash and Rhino have a strong correlation ($>$0.70) and WordPress, Glibc, PostgreSQL, and Firebug have a very strong correlation (Pearson's correlation test $>$0.90) between commit contribution and negative sentiments. We do not find a strong correlation between commit contribution and neutral sentiments.

RQ5: How has sentiment in the commit logs evolved over the period of time?

In this research question, our aim is to analyze the evolution of sentiments across time along with the number of commits made by developers. To achieve this goal all selected projects are considered. We group the sentiments (Positive, Negative, and Neutral) by each individual year to show how sentiments change across years along with the number of commits made by developers. Figure 7 shows the evolution of sentiments along with the number of commits across the years.

There is an increase in the neutral sentiment over the period of time in all the projects, which is a good sign for technical communication. Also, negative sentiment has decreased. Positive sentiment has stayed at the bottom throughout with small variations. Looking at the commit activity along with the sentiment evolution, it is evident that there is no relation between change in commit activity and sentiment evolution. One can observe a high percentage of negative sentiment irrespective of whether commit activity is high or low (as throughout in PostgreSQL and GNUCash, or Glibc from 1994 to 2009). On the other hand, negative sentiment remains low when commit activity is high in case of Eclipse, WordPress, Firebug, and Rhino.

Positive sentiment is the least kind of expression in the commit logs. There is more or less interplay between negative and neutral sentiments in all the projects. When neutral sentiment decreases, negative sentiment replaces it. So we can say developers are either negative or neutral while expressing themselves in commit logs. It is good to see a trend of improvement in neutral sentiment over the period of time.

Moreover, we also perceive that in most of the projects (PostgreSQL, WordPress, Eclipse-CDT, Firebug, and Rhino) the sentiments seem to be more positive in the starting years as compared to the ending years of observation. The reason for it could be that when the project is in its initial stage, it is less complex, having few issues. But as the project progresses it becomes more complex, more developers join the project with time and more issues need to be resolved.

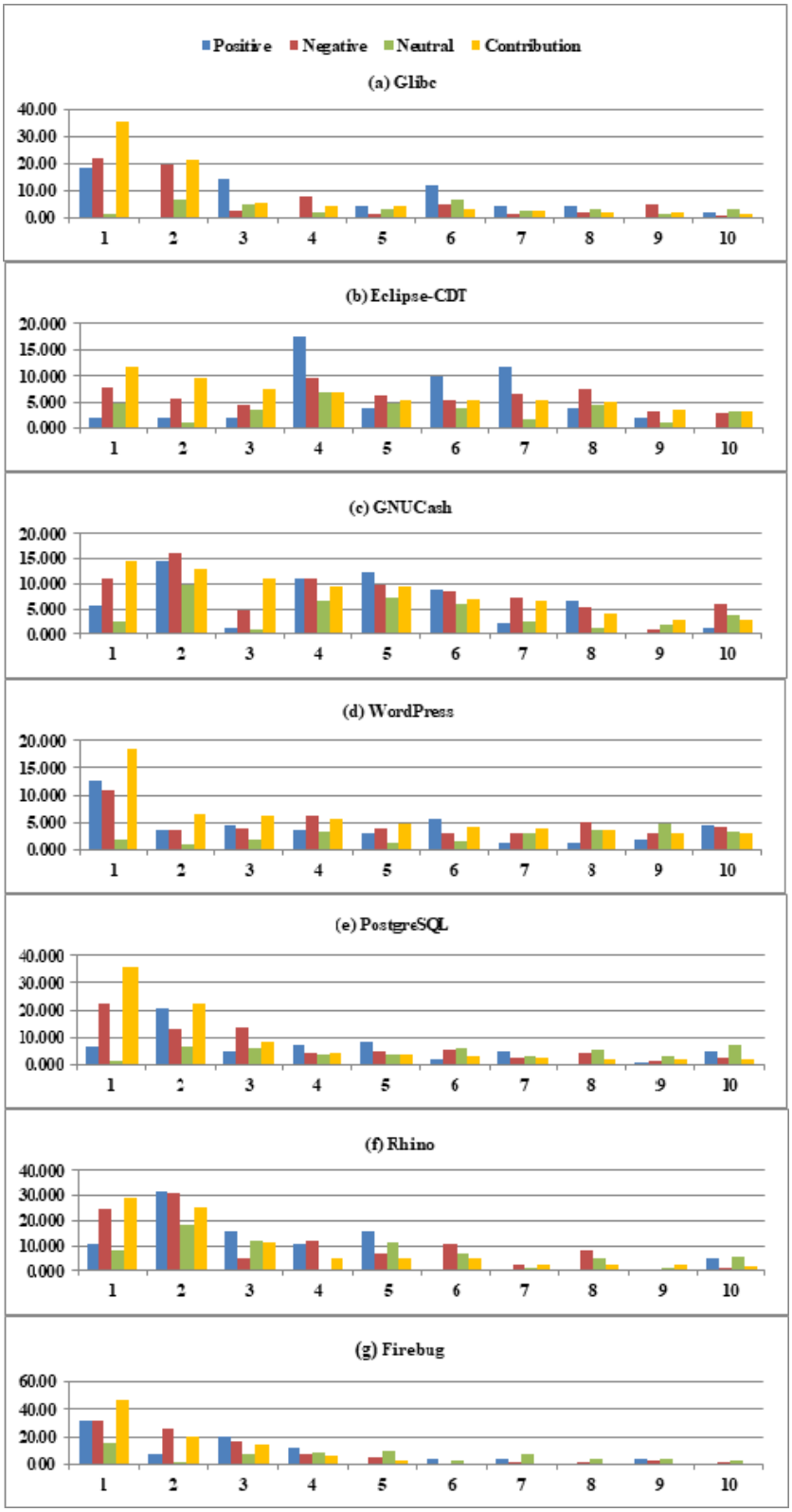


Figure 6. Commit contribution and sentiment volume:
(a) Glibc, (b) Eclipse-CDT, (c) GNUcash, (d) WordPress,
(e) PostgreSQL, (f) Rhino, and (g) Firebug

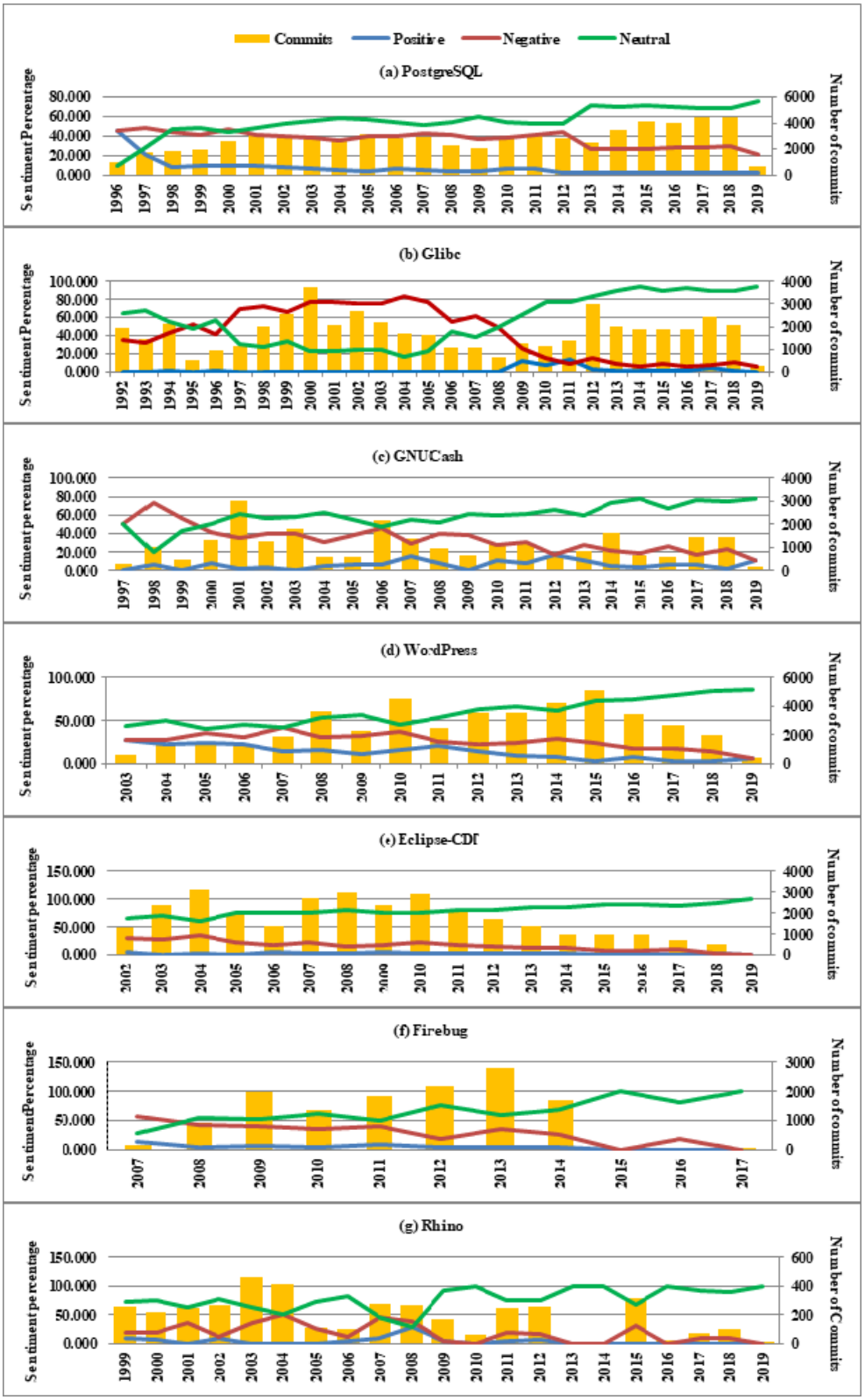


Figure 7. Evolution of sentiments by year:
(a) PostgreSQL, (b) Glibc, (c) GNUcash, (d) WordPress, (e) Eclipse-CDT, (f) Firebug, (g) Rhino

5. Discussion

In this work, we have examined 86,515 commit messages of seven well known GitHub projects to analyze the sentiments expressed by developers in the commit logs. Our main objective was to investigate the relation of team size, type of change activity, and commit contribution with sentiments in the commit logs. In addition to this, we also look into the evolution of sentiment in these projects. We found that most of the projects had high neutral sentiments in comparison to negative and positive ones. PostgreSQL indicates more negativity (36.03%) in sentiments and most of the negativity in the commit logs can be attributed to leading code contributors.

The majority of the commits in the commit logs are neutral. Our findings revealed that the team size of a project, type of change activity, and developers' commit contribution have an impact on the sentiment expressed in the commit logs. Furthermore, during the evolution of the project sentiments have different trends. We noticed that the commit logs have more positive/negative and less neutral sentiments in the initial years of the project in comparison to later years. The main reason behind it may be that in the starting years, a project is less complex and have a small number of issues but as the project progresses, more issues need to be resolved and large size of a team makes it a more formal platform and developers express themselves in a neutral way. Noticing the trend in Figure 7(b) indicates that expression in the glibc project, prior to 1990, was positive. It started getting negative after that. Age or maturity of a project does not influence developer sentiment expression in commit logs. But taking 2008 as the reference point, when Github was launched and most of the projects might have shifted to Github then, negative sentiment has decreased over the period of time. So it may be due to availability of the commit logs in the public domain, that sentiment expression has become more positive.

The study presented by Sinha [5] also examined the developers' sentiments in the commit logs. They identified that the majority of GitHub commits (74.74%) have neutral sentiments. As we compare our findings of RQ1, with results presented by Sinha [5], we noticed that our work found similar results. We observed that most of the commits in the commit log had a neutral sentiment. To compare our results with Sinha [5], we combined the sentiment results of all observed projects and found that in our analysis percentage of positive, negative, and neutral sentiments are 4.73%, 26.98%, 68.29%, respectively. In the case of our analysis positive sentiments are 2.47% and neutral sentiments are 6.45% less than Sinha's study. Negative sentiments are 8.93% higher than Sinha's study. This analysis shows that this result is very similar for a dataset different from the one studied in this research. They started with 28,466 OSS projects but considered only 5 projects for an in-depth sentiment analysis. So far detailed analysis, more work in this direction is required to confirm the findings for OSS projects of different domains and different sizes.

After this analysis, some actionable advice for the OSS community can be as follows:

- A project, large or small, should have a code of conduct mentioning the desired contribution quality in commit logs.
- In the issue tracking system, issues involving complex changes should be decomposed into multiple simple issues involving only two activities i.e. modify activity should be clubbed with either add or delete activity.
- Lead developers need to be aware of their sentiment expressions.
- Developers, looking for projects to contribute, can expect better commit logs, from sentiments point of view, in mature projects.

6. Threat to validity

The authors examined developers' sentiments in subject line of commit message but body of commit message may have different sentiments. For example, subject line may be neutral, but message body may be negative or vice versa. This aspect is missing in this study.

Same developer may have registered with multiple names. Multiple aliases related to same developers is not resolved that may influence the findings.

Moreover, the selection of the projects is biased as we included the projects having a valid Git repository while projects hosted on other platforms like Gitlab and Bitbucket are not taken into consideration. A subset of the research questions explored on a large Github dataset in [5] also gives results similar to the ones obtained here. In the future, we will extend our dataset to include more projects that are hosted on other software repositories. Furthermore, the result presented in this study only applies to OSS projects. In RQ4, the authors included the data of the top ten developers with very high commit activity while developers with very low commit (commit activity less than 1%) activities are not included in our analysis. In the future, we will extend our study to include developers with low levels of contribution.

Also combinations of file change viz. add + modify file, delete + modify file and add + delete + modify file are considered by authors to conduct analysis whereas impact of individual file change (add, delete, and modify) is not explored. Further research is required to examine the impact of individual file change like addition, deletion, and modification on sentiments.

Another limitation of our study is that we considered only a few factors to study the impact of developers' sentiments while there are many other factors such as code quality, gender, project age, and popularity that may influence sentiments expressed in the commit logs.

7. Conclusions

In this paper, the authors have analyzed the developers' sentiments in the commit logs of OSS projects. We examined 86,515 commit messages of the seven most popular OSS projects to analyze the sentiments expressed by developers in the commit logs. The authors investigated the impact of team size of the project, type of change activity (Type-1, Type-2, and Type-3) performed by developers, and code contribution volume to the sentiments expressed in the commit logs. Moreover, we analyzed the evolution of sentiments across years with respect to the number of commits made in each year.

Our study reveals that the majority of projects have neutral sentiments. This indicates that while creating commit log messages developers are more neutral. But when we compared negative with positive sentiments, we found that in case of three projects, percentage of negative sentiment is more than 10% greater than positive in all the projects, and negative sentiment is more than 23% higher than positive in four projects. In this study, we perceived that sentiments in the commit logs are influenced by team size. Neutral expressions are high with large team size and negative expressions are high with small team size. The type of change activity performed by developers also influences their sentiments expressed in the commit logs. Type-3 activity involving all the three change actions of addition, deletion, and modification, indicates more negative sentiments and low neutral sentiments. Furthermore, we also noticed that contribution size also impacts the volume of sentiment.

The developers with large commit contributions have more negativity in sentiments and developers having small commit contributions express more positive sentiments in commit logs. Besides, sentiments show different trend across years with respect to the number of commits made by developers. The developers have more positive sentiment in the initial years in comparison to the ending years. The neutral expression has increased over the period of time.

Our study results provide an understanding regarding developers' sentiments related to various software development team and project related concerns such as team size, contributor role, task complexity, and project evolution that will be helpful for OSS community in developing strategies to improve developer productivity and retention.

In the future, we intend to expand our research work by including more projects hosted on other platforms such as GitLab and Bitbucket. Large data sets and the complex interplay of various variables in this context demand to employ machine learning or deep learning techniques to identify the association.

We also want to look into why expression in small teams is negative and explore it from the perspectives of informal interactions as well as work pressure. This study can also be extended to include specific type of developers, e.g., lead or occasional, to study the difference in their sentiment expressions in the commit logs.

Acknowledgments

The research work presented in this paper is sponsored by UGC, Government of India. The authors are appreciative of UGC to provide funding under Rajiv Gandhi National Fellowship scheme to the first author. The authors are also grateful to the Department of Computer Science, Guru Nanak Dev University Amritsar, and Punjab for infrastructure and scholastic aid towards the ongoing research.

References

[1] D. Graziotin and F. Fagerholm, "Happiness and the productivity of software engineers," in *Rethinking Productivity in Software Engineering*. Springer, 2019, pp. 109–124.

[2] M. De Choudhury and S. Counts, "Understanding affect in the workplace via social media," in *Proceedings of the Conference on Computer Supported Cooperative Work*, 2013, pp. 303–316.

[3] B. Liu et al., "Sentiment analysis and subjectivity," *Handbook of Natural Language Processing*, Vol. 2, No. 2010, 2010, pp. 627–666.

[4] E. Guzman, D. Azócar, and Y. Li, "Sentiment analysis of commit comments in GitHub: An empirical study," in *Proceedings of the 11th Working Conference on Mining Software Repositories*, 2014, pp. 352–355.

[5] V. Sinha, A. Lazar, and B. Sharif, "Analyzing developer sentiment in commit logs," in *Proceedings of the 13th International Conference on Mining Software Repositories*, 2016, pp. 520–523.

[6] N. Singh and P. Singh, "How do code refactoring activities impact software developers' sentiments? – An empirical investigation into GitHub commits," in *24th Asia-Pacific Software Engineering Conference (APSEC)*. IEEE, 2017, pp. 648–653.

[7] M.R. Islam and M.F. Zibran, "Sentiment analysis of software bug related commit messages," *Network*, Vol. 740, 2018, p. 740.

[8] P. Tourani, Y. Jiang, and B. Adams, "Monitoring sentiment in open source mailing lists: exploratory study on the apache ecosystem," in *Proceedings of 24th Annual International Conference on Computer Science and Software Engineering*, Vol. 14, 2014, pp. 34–44.

[9] J. Ding, H. Sun, X. Wang, and X. Liu, "Entity-level sentiment analysis of issue comments," in *Proceedings of the 3rd International Workshop on Emotion Awareness in Software Engineering*, 2018, pp. 7–13.

[10] F. Jurado and P. Rodriguez, "Sentiment analysis in monitoring software development processes: An exploratory case study on GitHub's project issues," *Journal of Systems and Software*, Vol. 104, 2015, pp. 82–89.

[11] R. Paul, A. Bosu, and K.Z. Sultana, "Expressions of sentiments during code reviews: Male vs. female," in *26th International Conference on Software Analysis, Evolution and Reengineering (SANER)*. IEEE, 2019, pp. 26–37.

[12] D. Garcia, M.S. Zanetti, and F. Schweitzer, "The role of emotions in contributors activity: A case study on the gentoo community," in *International Conference on Cloud and Green Computing*. IEEE, 2013, pp. 410–417.

[13] M.R. Islam and M.F. Zibran, "Exploration and exploitation of developers' sentimental variations in software engineering," in *Research Anthology on Recent Trends, Tools, and Implications of Computer Programming*. IGI Global, 2021, pp. 1889–1910.

[14] A. Murgia, P. Tourani, B. Adams, and M. Ortu, "Do developers feel emotions? An exploratory analysis of emotions in software artifacts," in *Proceedings of the 11th Working Conference on Mining Software Repositories*, 2014, pp. 262–271.

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

[16] N. Novielli, F. Calefato, F. Lanubile, and A. Serebrenik, "Assessment of off-the-shelf SE-specific sentiment analysis tools: An extended replication study," *Empirical Software Engineering*, Vol. 26, No. 4, 2021, pp. 1–29.

[17] K. Sun, H. Gao, H. Kuang, X. Ma, G. Rong et al., "Exploiting the unique expression for improved sentiment analysis in software engineering text," *arXiv preprint arXiv:2103.13154*, 2021.

[18] E. Biswas, M.E. Karabulut, L. Pollock, and K. Vijay-Shanker, "Achieving reliable sentiment analysis in the software engineering domain using BERT," in *International Conference on Software Maintenance and Evolution (ICSME)*. IEEE, 2020, pp. 162–173.

[19] N. Novielli, F. Calefato, D. Dongiovanni, D. Girardi, and 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*, 2020, pp. 158–168.

[20] M.R. Wrobel, "The impact of lexicon adaptation on the emotion mining from software engineering artifacts," *IEEE Access*, Vol. 8, 2020, pp. 48 742–48 751.

[21] M. Obaidi and J. Klünder, "Development and application of sentiment analysis tools in software engineering: A systematic literature review," *Evaluation and Assessment in Software Engineering*, 2021, pp. 80–89.

[22] S.F. Huq, A.Z. Sadiq, and K. Sakib, "Is developer sentiment related to software bugs: An exploratory study on GitHub commits," in *27th International Conference on Software Analysis, Evolution and Reengineering (SANER)*. IEEE, 2020, pp. 527–531.

[23] R. Kaur and K.K. Chahal, "Analysis of developers' sentiments in commit comments," in *International Conference on Advanced Informatics for Computing Research*. Springer, 2020, pp. 3–12.

[24] S. Bharti and H. Singh, "Investigating developers' sentiments associated with software cloning practices," in *International Conference on Advanced Informatics for Computing Research*. Springer, 2018, pp. 397–406.

[25] R. Souza and B. Silva, "Sentiment analysis of Travis CI builds," in *14th International Conference on Mining Software Repositories (MSR)*. IEEE, 2017, pp. 459–462.

[26] D. Pletea, B. Vasilescu, and A. Serebrenik, "Security and emotion: sentiment analysis of security discussions on GitHub," in *Proceedings of the 11th Working Conference on Mining Software Repositories*, 2014, pp. 348–351.

[27] I.A. Khan, W.P. Brinkman, and R.M. Hierons, "Do moods affect programmers' debug performance?" *Cognition, Technology and Work*, Vol. 13, No. 4, 2011, pp. 245–258.

[28] S.C. Müller and T. Fritz, "Stuck and frustrated or in flow and happy: Sensing developers' emotions and progress," in *37th International Conference on Software Engineering*, Vol. 1. IEEE, 2015, pp. 688–699.

[29] D. Graziotin, X. Wang, and P. Abrahamsson, "Happy software developers solve problems better: Psychological measurements in empirical software engineering," *PeerJ*, Vol. 2, 2014, p. e289.

[30] M.R. Wrobel, "Emotions in the software development process," in *6th International Conference on Human System Interactions (HSI)*. IEEE, 2013, pp. 518–523.

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

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

[33] M. Ortu, A. Murgia, G. Destefanis, P. Tourani, R. Tonelli et al., "The emotional side of software developers in JIRA," in *13th Working Conference on Mining Software Repositories (MSR)*. IEEE, 2016, pp. 480–483.

[34] M.M. Rahman, C.K. Roy, and I. Keivanloo, "Recommending insightful comments for source code using crowdsourced knowledge," in *15th International Working Conference on Source Code Analysis and Manipulation (SCAM)*. IEEE, 2015, pp. 81–90.

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

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

[37] R. Jongeling, S. Datta, and A. Serebrenik, "Choosing your weapons: On sentiment analysis tools for software engineering research," in *International Conference on Software Maintenance and Evolution (ICSME)*. IEEE, 2015, pp. 531–535.

[38] K. Beecher, C. Boldyreff, A. Capiluppi, and S. Rank, "Evolutionary success of open source software: An investigation into exogenous drivers," *Electronic Communications of the EASST*, 2008.