Towards Understanding and Exploiting Developers' Emotional Variations in Software Engineering

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

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

***Abstract*—Software development is highly dependent on human efforts and collaborations, which are immensely affected by emotions. This paper presents a quantitative empirical study of the emotional variations in different types of development activities (e.g., bug-fixing tasks) and development periods (i.e., days and times), in addition to in-depth investigation of emotions' impacts on software artifacts (i.e., commit messages) and exploration of scopes for exploiting emotional variations in software engineering activities. We study emotions in more than 490 thousand commit comments across 50 open-source projects. The findings add to our understanding of the role of emotions in software development, and expose scopes for exploitation of emotional awareness in improved task assignments and collaborations.**

I. Introduction

Emotions are inseparable part of human nature, which influence people's activities and interactions, and thus emotions affect task quality, productivity, creativity, group rapport and job satisfaction [2], [8], [21]. Software development, being highly dependent on human efforts and interactions, is more susceptible to emotions of the practitioners. Hence, a good understanding of the developers' emotions and their influencing factors can be exploited for effective collaborations, task assignments [6], and in devising measures to boost up job satisfaction, which, in turn, can result in increased productivity and projects' success [4].

Several studies have been performed in the past for understanding the role of human aspects on software development and engineering. Some of those earlier studies address *when* and *why* employees get affected by emotions [2], [12], [13], [23], [27], whereas some other work address *how* [10], [14], [15], [20], [28] the emotions impact the employees' performance at work. Despite those earlier attempts, software engineering practices still lack theories and methodologies for addressing human factors such as, emotions, moods and feelings [11], [13]. Hence, the community calls for research on the role of emotions in software engineering [14], [21], [25].

Some software companies try to capture the developers' emotional attachments to their jobs by means of traditional approaches such as interviews and surveys [28]. Capturing emotions with the traditional approaches is more challenging for projects relying on geographically distributed team settings and voluntary contributions (e.g., open-source projects) [5], [12]. Thus, to supplement or complement those traditional sources, software artifacts such as the developers' commit comments/messages have been identified for the extraction of important information including developers' emotional states [12], [13], [23].

In this work, we study the polarity (i.e., positivity, negativity, and neutrality) of emotions expressed in commit messages as posted by developers contributing to open-source projects. In particular, we address the following four research questions.

RQ1: *Do developers express different levels (e.g., high, low) and polarity (i.e., positivity, negativity, and neutrality) of emotions when they commit different types (e.g., bug-fixing, new feature implementation, refactoring, and dealing with energy related concerns) of development tasks?*
— If we can distinguish development tasks at which the developers express high negative emotions, low positive emotions, or an overall low emotional involvements, stipulating measures can be introduced to emotionally influence the emotions of the developers working on those particular types of development tasks resulting in higher success rate.

RQ2: *Can we distinguish a group of developers who express more emotions (positive or negative) in committing a particular type (e.g., bug-fixing) of tasks?*
— Programmers who develop in them positive emotions while carrying out a given development task can be more efficient and quicker in completing the task [20] resulting in reduced software cost. Thus, distinguishing a group of practitioners having positive emotional attachment to a particular task can be useful in effective task assignments.

RQ3: *Do the developers' polarity (i.e., positivity, negativity, and neutrality) of emotions vary in different days of a week and in different times of a day*?
— If we can identify any particular days and times when developers express significant negative emotions, then managers can take motivating steps to boost up the developers positive feelings on those days and times. Guzman et al. [12] reported that commit comments posted on Mondays tend to have more negative emotions. We also want to verify their claim using a substantially larger data-set.

RQ4: *Do the developers' emotions have any impact on the lengths of commit comments they write*?
— Commit messages are pragmatic means of communication among the developers contributing to the same project. Ideally, commit comments contain important information about the underlying development tasks, and the length of developers'

978-1-5090-0809-4/16/$31.00 copyright 2016 IEEE
SERA 2016, June 8-10, 2016, Baltimore, USA

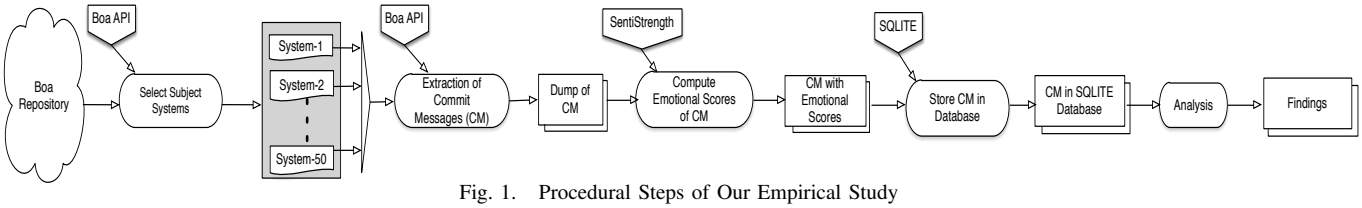

Fig. 1. Procedural Steps of Our Empirical Study

work description is an indication of the description quality [16]. If any relationship can be found between the developers' emotional state and the lengths of commit comments, then project managers can take steps to stimulate the developers emotional states to get high quality commit comments containing enough contextual information.

II. Methodology

To address the aforementioned research questions, we extract emotions from the developers' commit messages using `SentiStrength` [26], which is a state-of-the-art sentiment analysis tool. `SentiStrength` was previously used for similar purposes [9], [13], [27] and was reported to be good candidate for analyzing emotions in commit comments [12]. In the following subsections, we first briefly introduce sentiment analysis with `SentiStrength` (Section II-A) and then, we describe the metrics (Section II-B), tuning of `SentiStrength` (Section II-C) for software engineering context, and data collection approaches (Section II-D) used in our study. The procedural steps of our empirical study are summarized in Figure 1.

A. Sentiment Analysis

Sentiment analysis using `SentiStrength` on a given piece of text (e.g., a commit message) c computes a pair $\langle \rho_c, \eta_c \rangle$ of integers, where $+1 \leq \rho_c \leq +5$ and $-5 \leq \eta_c \leq -1$. Here, ρ_c and η_c respectively represent the positive and negative emotional scores for the given text c.

A given text c is considered to have positive emotions if $\rho_c > +1$. Similarly, a text is held containing negative emotions when $\eta_c < -1$. Note that, a given text can exhibit both positive and negative emotions at the same time, and a text is considered emotionally neutral when the emotional scores for the text appear to be $\langle 1, -1 \rangle$. Further details about the sentiment analysis algorithm of `SentiStrength` and the interpretation of its outputs can be found elsewhere [26].

B. Metrics

To carry out our analyses for deriving the answers to the research questions, we formulate the following metrics. Given a set $\mathcal{C}$ of commit messages, we can obtain two subsets $\mathcal{C}+$ and $\mathcal{C}_-$ defined as follows:
$\mathcal{C}_+ = \{c \quad | c \in \mathcal{C}, \rho_c > +1\}$ and $\mathcal{C}_- = \{c \quad | c \in \mathcal{C}, \eta_c < -1\}$.

Mean Positive Emotional Score for a set $\mathcal{C}$ of commit messages, denoted as $\mathcal{P}(\mathcal{C})$, is defined as:

$$\mathcal{P}(\mathcal{C}) = \frac{\sum_{c \in \mathcal{C}_+} \rho_c}{|\mathcal{C}_+|} \tag{1}$$

Mean Negative Emotional Score for a set $\mathcal{C}$ of commit comments, denoted as $\mathcal{N}(\mathcal{C})$, is defined as follows:

$$\mathcal{N}(\mathcal{C}) = \frac{\sum_{c \in \mathcal{C}_-} |\eta_c|}{|\mathcal{C}_-|} \tag{2}$$

Cumulative Emotional Score for a particular commit message c, denoted as $\mathcal{T}(c)$, is defined as follows,

$$\mathcal{T}(c) = \rho'_c + \eta'_c \tag{3}$$

where,

$$\rho'_c = \begin{cases} \rho_c, & \text{if } \rho_c > +1. \\ 0, & \text{otherwise.} \end{cases} \qquad \eta'_c = \begin{cases} |\eta_c|, & \text{if } \eta_c < -1. \\ 0, & \text{otherwise.} \end{cases}$$

C. Tuning of SentiStrength

The sentiment analysis tool `SentiStrength` was reported to have 60.7% precision for positive texts and 64.3% for negative texts [26]. To the best of our knowledge, all such sentiment analysis tools including `SentiStrength` are highly dependent on the polarities of individual words in a given text in computation of its emotional scores. `SentiStrength` was originally trained on documents on the social web. In a technical field such as software engineering, commit messages include many keywords which have polarities in terms of dictionary meanings, but do not really express any emotions in their technical context. For example, 'Super', 'Support', 'Value' and 'Resolve' are English words with known positive emotions, while 'Dead', 'Block', 'Default', and 'Garbage' are known to have negative emotions, but neither of these words really bear any emotions in software development artifacts. Those are simply some domain specific technical terms with especial contextual meanings.

To save `SentiStrength`'s computation of emotional scores from being mislead by such technical terms, we tune the tool for application in our software engineering context. Based on our manual investigation, experience, and literature review [23], [27], we identify a total of 49 terms, which can be misinterpreted by `SentiStrength`. These misleading terms are: *'Arbitrary', 'Block', 'Bug', 'Conflict', 'Constraint', 'Corrupt', 'Critical', 'Dynamic', 'Dead', 'Death', 'Default', 'Defect', 'Defensive', 'Disabled', 'Eliminate', 'Error', 'Exceptions', 'Execute', 'Failure', 'Fatal', 'Fault', 'Force', 'Garbage', 'Greater', 'Inconsistency', 'Interrupt', 'Kill', 'Like', 'Obsolete', 'Pretty', 'Redundant', 'Refresh', 'Regress', 'Resolve', 'Restrict', 'Revert', 'Safe', 'Security', 'Static', 'Super', 'Support', 'Success', 'Temporary', 'Undo', 'Value', 'Violation', 'Void', 'Vulnerable' and 'Wrong'*.

SentiStrength provides the flexibility to modify its existing lexicons' emotional interpretation to customize it for a target context (i.e., software engineering, in this work). For our purpose, we neutralize SentiStrength's interpretation of the aforementioned technical jargons, as such was also suggested in earlier studies in the area [23], [27].

Having SentiStrength tuned according to the procedure described above, we manually verify the impact of the tuning using a random sample of 200 commit messages extracted from Boa [7], and we found a 26% increase of precision (checked by comparing SentiStrength's computation of emotional polarities with subjective human interpretation over each of the 200 commit messages). Thus, for our work, we use this improved instance of SentiStrength tuned for use in software engineering context.

D. Data Collection

We study commit messages for open-source projects obtained through Boa [7]. Boa is a recently introduced infrastructure with a domain specific language and public APIs to facilitate mining software repositories. We use the largest (as of February 2016) data-set from Boa, which is categorized as "full (100%)" and consists of more than 7.8 million projects collected from GitHub before September 2015.

From this large data-set, we select the top 50 projects having the highest number of commits. We study all the commit messages in these projects, which constitute 490,659 commit comments. Associated information such as, committers, commit timestamps, types of underlying work, revisions and project IDs are kept in a local relational database for convenient access and query. For each of the commit messages, we compute the emotional scores using the tuned SentiStrength tool. Table I shows some examples of emotional and neutral commit comments in our dataset and computation of their emotional scores.

III. Analysis and Findings

The research questions *RQ1, RQ2, RQ3* and *RQ4* are respectively addressed in Section III-A, Section III-B, Section III-C, and in Section III-D.

A. Emotional Variations in Different Task Types

We investigate whether developers' emotions vary based on their involvements in four different types of software development tasks: (a) bug-fixing tasks, (b) new feature implementation, (c) refactoring, and (d) energy-aware development. We consider that the first three types of tasks mentioned above are self-explanatory. The fourth one (i.e., energy-aware development) deals with software issues with consumption of energy, measured in terms of usage of resources such as processing power and memory. Energy-aware development is a recent important topic in the area of green computing research. Categorization of development tasks in this manner are also found in earlier studies [1], [3], [22] in software engineering research.

Task-based Characterization of Commits: To distinguish commits dealing with *bug-fixing* tasks, we rely on Boa's public APIs, which readily indicate whether a commit message is associated with bug-fixing task, or not.

To identify *energy-aware* commit messages, we select a list of keywords and search those keywords in commit messages. A commit message will be considered as energy-aware commit, if the commit message contains any of the selected keywords. The identified keywords are: **energy consum*, *energy efficien*, *energy sav*, *save energy*, *power consum*, *power ecien*, *power sa*, *save power*, *energy drain*, *energy leak*, *tail energy*, *power efficien*, *high CPU*, *power aware*, *drain*, *no sleep*, *battery life* and *battery consum**. The character '*' in each keyword works as a wildcard, i.e., a query will select those commits messages, which contain at least one of these keywords, regardless of the beginning or the end of the commit message. Note that, these keywords were also used in earlier studies [3], [17], [19], [22] for similar purposes.

To recognize commit messages dealing with *new feature implementation* and *refactoring* tasks, we select those keywords, which were used by Ayalew and Mguniin [1] in their work. Keywords **add* and *new feature** are used to categorize commit messages, which are related to new feature development. And **refactor* and *code clean** keywords are used to distinguish those commit messages, which are posted by developers during code refactoring tasks. Note that, a developer may perform refactoring while fixing a bug. Thus, a commit message can be characterized relevant to more than one categories of tasks.

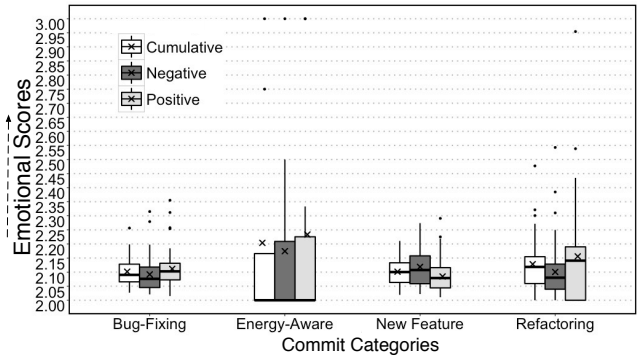

Fig. 2. Distribution of mean positive, negative, and cumulative emotional scores in commits messages dealing with different types of tasks

Investigation: The numbers of commit messages found relevant to each of the four categories of development tasks are presented in the second column from left in Table II. The boxplot in Figure 2 presents the distribution of mean positive, negative, and cumulative emotional scores in each type of task for each of the 50 projects. An 'x' mark in a box in the boxplot indicates the mean emotional scores over all the projects.

As observed in Figure 2, emotional scores (positive, negative and cumulative) for energy-aware commit messages are much higher than those in commit messages for three other tasks, and there is not much variations in the emotional scores

TABLE I
EXAMPLES OF COMMIT COMMENTS AND COMPUTATION OF THEIR EMOTIONAL SCORES

Boa Proj. ID	Commit/Revision ID: Commit Comment (c)	Emo. Score		
		ρ_c	η_c	$\mathcal{T}(c)$
12562083	7519717434bb0ae5fad5329885bd184e7b502d27: Fixes #1721 Committing work by Arnfried (EXCELLENT!)	5	-1	5
689344	2605951f8b73a963beb01b3806b3ad43ce638848: Don't save mute setting. Extremely annoying to start with lack of audio and have no idea what causes it	1	-5	5
11814891	01845191185d0a14960f1542ac77f512f8749514: a bit more detailed test; hope this avoids some reflection searches in FF emulation and makes the monster faster in special situations	3	-2	5
689344	00058618c9c3f1f9fc4d9310012a0d1881c0c940: (RMenu) RMenu refactor - have function pointers for menu struct	1	-1	0

TABLE II
COMMITS OF DIFFERENT TASK CATEGORIES AND *MWW* TESTS BETWEEN POSITIVE AND NEGATIVE EMOTIONS IN THEM

Task Categories	# of Commits	P-value	Significant?
Bug-Fixing	117,249	0.03288	Yes ($P < \alpha$)
New Feature	89,019	0.00256	Yes ($P < \alpha$)
Refactoring	5,431	0.04006	Yes ($P < \alpha$)
Energy-Aware	182	0.39743	No ($P > \alpha$)

among these three tasks. To verify the statistical significance of these observations, we conduct *Mann-Whitney-Wilcoxon (MWW)* tests [24] (with $\alpha = 0.05$) between the distributions of mean cumulative emotional scores in commit messages for each possible pair of development tasks. The results of the *MWW* tests are presented in Table III. The P-values reported by the tests, as compared with α, suggest statistical significance of our observations.

TABLE III
MWW TESTS BETWEEN CUMULATIVE EMOTIONAL SCORES OF COMMIT MESSAGES DEALING WITH DIFFERENT TYPES OF TASKS

Task Categories	Bug-Fixing	Refactoring	New Feature	Energy-Aware
Bug-Fixing	-	0.75656	0.89656	0
Refactoring	0.75656	-	0.71884	0
New Feature	0.89656	0.71884	-	0
Energy-Aware	0	0	0	-

Again, looking at Figure 2, we see that the commit messages, which are posted during the new features implementation tasks, show more negative emotions than positive ones. Opposite observations are evident for commit messages for three other types of tasks. To verify the statistical significance of our observations in the variations of polarity (positivity and negativity) of emotions, for each of the four types of development tasks, we separately conduct *MWW* tests between the mean positive and negative emotional scores of commit messages. The results of the *MWW* tests are presented in the right-most two columns in Table II. The P-values of tests, as compared with α, suggest statistical significance of our observations for bug-fixing, new feature implementation and refactoring tasks, but not for the *energy-aware* development tasks.

Based on our observations and statistical tests, we derive the answer to the research question *RQ1* as follows:

Ans. to RQ1: *Developers express significantly high positive and negative emotions almost equally in committing energy-aware tasks. For bug-fixing and refactoring tasks, positive emotions are significantly higher than negative emotions. And surprisingly, for new feature implementation tasks, negative emotions are significantly higher than positive polarity.*

B. Emotional Variations in Bug-Fixing Tasks

It is natural that different developers have different expertise, comfort-zones, and interests with respect to types of tasks. The research question *RQ2* addresses the possibility of distinguishing a set of developers who particularly express positive emotions at the particular type of task at hand. In addressing the research question *RQ2*, we choose the *bug-fixing* tasks as a representative to any particular type of tasks and continue as such.

Across all the projects, we distinguish 20 developers, who are the authors of the bug-fixing commit messages having the highest positive mean emotional scores. Let $\mathcal{D}_p$ denote the set of these 20 developers. Similarly, we form another set $\mathcal{D}_n$ consisting of 20 developers, who are the authors of bug-fixing commit comments having the highest negative mean emotional scores. By the union of these two sets, we obtain a set $\mathcal{D}$ of 30 developers who are authors of bug-fixing commits with the highest mean positive or negative emotional scores. Mathematically, $\mathcal{D} = \mathcal{D}_p \cup \mathcal{D}_n$.

These 30 developers are the authors of 112,462 commits messages among which 32,088 are bug-fixing commits. For each of these 30 developers, we compute a ratio $\mathcal{R}(d)$ as follows:

$$\mathcal{R}(d) = \frac{\mathcal{P}(\mathcal{C}_d)}{\mathcal{N}(\mathcal{C}_d)},\ where,\ d \in \mathcal{D} \tag{4}$$

Here, $\mathcal{C}_d$ denotes the set of bug-fixing commit comments posted by developer d. Notice that, for a particular developer d, the ratio $\mathcal{R}(d)$ close to 1.0 indicates that the positive and negative emotions are almost equal for the developer d. If $\mathcal{R}(d)$ is much higher than 1.0, the developer d shows more positive emotions at bug-fixing tasks compared to negative emotions. The opposite holds when $\mathcal{R}(d)$ is much lower than 1.0. However, a threshold scheme seems necessary to determine when the value of $\mathcal{R}(d)$ can be considered significantly close to or distant from 1.0.

Clustering Analysis: Instead of setting an arbitrary threshold, we apply unsupervised *Hierarchical Agglomerative Clustering* for partitioning the values of $\mathcal{R}(d)$. The dendrogram produced from this clustering is presented in Figure 3. In the dendrogram, we identify three major clusters/groups, two

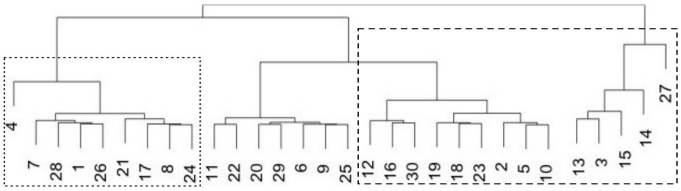

Fig. 3. Hierarchical agglomerative clustering of 30 developers enumerated as $1, 2, 3, \ldots, 30$

marked (by us) with dotted rectangles and the third left unmarked in the middle. This middle cluster, denoted as G_b, represents the set of those developers, who equally express positive and negative emotions during bug-fixing. We have, $0.992 \leq \mathcal{R}(d) \leq 1.0, \forall d \in G_b$.

The set of developers who are included in the right-most cluster exhibit more positive emotions compared to negative emotions during bug-fixing. Let G_p denote the cluster of these developers. Here, $1.005 \leq \mathcal{R}(d) \leq 1.178, \forall d \in G_p$. The set of developers who render more negative emotions towards bug-fixing belong to left-most cluster, denoted as G_n. We have, $0.919 \leq \mathcal{R}(d) \leq 0.982, \forall d \in G_n$.

TABLE IV
MWW TESTS OVER $\mathcal{R}(d)$ SCORES OF COMMIT MESSAGES WRITTEN BY DEVELOPERS IN EACH CLUSTER

Cluster	G_p	G_n	G_b
P**-values**	0.00798	0.0268	0.26109
Significant?	Yes ($P < \alpha$)	Yes ($P < \alpha$)	No ($P > \alpha$)

Statistical Significance: For each of the three clusters, we separately conduct *MWW* tests between the mean positive and negative emotional scores of the commit messages to verify the statistical significances of their differences. The results of the separate *MWW* tests (with $\alpha = 0.05$) over each of the clusters are presented in Table IV. The P-values in Table IV indicate statistical significance in the differences in positive and negative emotions for clusters G_p and G_n. As expected, no such significant difference found for the cluster G_b as in this cluster, positive and negative emotions are expressed equally. Thus, our clustering of the developers appears to be accurate with statistical significance. Hence, we answer the research question *RQ2* as follows:

Ans. to RQ2: *We have been able to distinguish sets of developers who show either high positive or high negative emotions in bug-fixing commit messages while some other developers are found to express both positive and negative emotions almost equally. The same approach can be applied to distinguish such groups of developers for other types of development tasks.*

C. Emotional Variations in Days and Times

For each of the projects, we group all the commit messages into seven disjoint sets in accordance with the days of the week those are committed.

Figure 4 plots the average (over each project) positive, negative, and cumulative emotional scores in commit messages posted in different days in a week. Among *all the seven days* of

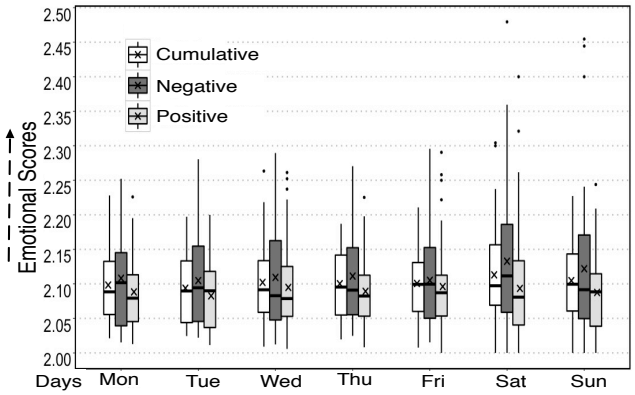

Fig. 4. Distribution of mean positive, negative, and cumulative emotional scores in commit comments posted in different days of week

TABLE V
MWW TESTS OVER CUMULATIVE EMOTIONAL SCORES OF COMMIT MESSAGES WRITTEN IN DIFFERENT DAYS OF WEEK

Day	Sat	Sun	Mon	Tue	Wed	Thu	Fri
Sat	-	0.44	0.23	0.11	0.33	0.41	0.35
Sun	0.44	-	0.71	0.42	0.77	0.98	0.84
Mon	0.23	0.71	-	0.68	0.79	0.71	0.84
Tue	0.11	0.42	0.68	-	0.55	0.41	0.49
Wed	0.33	0.77	0.79	0.55	-	0.83	0.96
Thu	0.41	0.98	0.71	0.41	0.83	-	0.86
Fri	0.35	0.84	0.84	0.49	0.96	0.86	-

a week, negative emotions appear to be slightly higher in commit messages posted during the weekends (i.e., Saturday and Sunday) than those posted in weekdays (i.e., Monday through Friday). Not much differences are visible in the emotional scores for commit messages posted in the *five weekdays*. *MWW* tests (with $\alpha = 0.05$) between the distributions of emotional scores in each possible pair of the days of a week suggest no statistical significance in the differences of emotions. P-values of the *MWW* tests are presented in Table V. As can be seen in Table V, for all values of P's, $\alpha < 0.11 \leq P$.

To study the relationship between developers emotions and times of a day when commit comments are posted, we divide the 24 hours of a day in three periods (a) 00 to 08 hours as *before working hours*, (b) 09 to 17 hours as regular *working hours* and (c) 18 to 23 hours as *after working hours*. Then for each project, we again organize the commit messages into three disjoint sets based on their timestamps of posting.

Figure 5 presents the mean (over each project) positive and negative emotional scores (computed using Equation 1 and Equation 2) in commit messages posted in these three periods. Again, in Figure 5, we do not see much variations in the emotional scores of commit messages posted at different periods. *MWW* tests (with $\alpha = 0.05$) between the distributions of mean positive and negative emotional scores in each possible pair of the periods indicate no statistical significance in their differences. P-values of the *MWW* tests are presented in Table VI. Hence, we derive the answer to the research question *RQ3* as follows:

Ans. to RQ3: *There is no significant variations in the developers' emotions in different times and days of a week.*

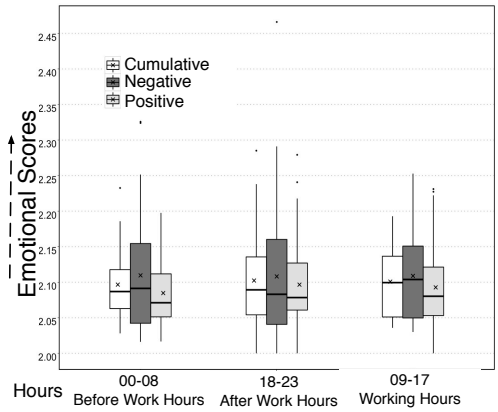

Fig. 5. Distribution of mean positive, negative, and cumulative emotional scores in commit comments posted in different periods of day

TABLE VI
MWW TESTS OVER CUMULATIVE EMOTIONAL SCORES OF COMMIT MESSAGES WRITTEN IN DIFFERENT TIMES OF A DAY

Hours in a Day	00-08	09-17	18-23
00-08	-	0.59612	0.84148
09-17	0.59612	-	0.85716
18-23	0.84148	0.85716	-

D. Emotional Impacts on Commit Lengths

To investigate the existence of any relationship between emotions and lengths of commit messages, across all the 50 projects, we distinguish 141,033 commit comments, which are one to 50 words in length having cumulative emotional scores (computed using Equation 3) higher than one. For each project, we organize these emotional commit messages into four disjoint groups based on their lengths as shown in Figure 6, which plots the mean (over each project) cumulative emotional scores of commit messages in the four groups. As seen in the figure, the emotional scores are strictly higher for the groups with lengthier commit messages.

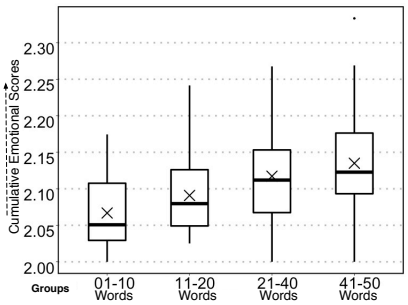

Fig. 6. Distribution of mean cumulative emotional scores of commit comments in groups of different lengths

Table VII presents the frequencies of commit messages in the four groups and having different cumulative emotional scores. A *Chi-squared* [24] test ($P = 2.2 \times 10^{-16}, \alpha = 0.05$) also strongly indicates statistical significance of the relationship between emotional scores and commit lengths. Next, we verify the significance of the *direction* of relationship (i.e., if one increases or decreases with the increase of another).

TABLE VII
NUMBER OF COMMIT MESSAGES WITH DIFFERENT LENGTHS (IN WORDS) AND CUMULATIVE EMOTIONAL SCORES

	Commit Length	**# of Commits Comments with $\mathcal{T}(c) =$**						
		02	**03**	**04**	**05**	**06**	**07**	**08**
Groups	**01-10**	46,486	2,734	2,558	245	28	12	3
	11-20	42,144	3,967	4,627	876	145	16	1
	21-40	22,732	2,633	5,008	1,155	203	31	2
	41-50	3,255	409	1,275	399	84	5	0

Fitting of a *Generalized Linear Model* [24] on the emotional score and length of every emotional commit message confirms (with $\beta = +0.01134, P = 2 \times 10^{-16}, \alpha = 0.001$) the positive correlation between emotional scores and commit lengths. Based on the analyses, we now derive the answer to the research question *RQ4* as follows:

Ans. to RQ4: *Developers' emotions have statistically significant impacts on the lengths of commit messages they write. Developers post longer commit comments when they are emotionally active.*

IV. THREATS TO VALIDITY

In this section, we discuss the limitations of our work, the threats to the validity of our findings, and our attempts to minimize those threats.

Internal Validity: The *internal validity* of our work depends on the accuracy of the tool's computation of emotional scores. `SentiStrength` was reported to be effective in sentiment analysis [26] and suitable for extraction of emotions from commit comments [12]. `SentiStrength` has relatively high accuracy compared to other tools of its kind and thus `SentiStrength` was used for sentiment analysis in earlier work in software engineering research [3], [9], [12], [13], [27]. Moreover, for use in our work, we increased it accuracy in emotion extraction by 26% through tuning the tool for application in software engineering context (Section II-C).

Nevertheless, the tuned tool is not 100% accurate in determining emotional polarities of commit messages, and it was not possible to perform manual sanity check by going through each of the 490,659 commit messages included in our work. We are aware of this threat, although we minimized it by contextual customization of `SentiStrength`.

Construct Validity: The choice of the 30 developers in examining the relationships between emotions and bug-fixing tasks (Section III-B) can be questioned. Note that, these 30 developers are the authors of more than *112 thousand* commit messages (22.85%), which is a large sample of data for dependable analysis. The objective was to check if it was possible to distinguish a group of developers who are emotionally more active towards a particular type of task. If we chose a fair number of developers other than our choice of 30, we would still be able to distinguish a set of target developers. In that case, the size of the set of developers might be different from what we found using the 30 developers, but this does not invalidate the findings of the work.

One may also question the validity of our categorization of the developers' commits in different days and periods

(Section III-C), considering the possibility that the projects and developers may be physically located at different geographic locations and time-zones. However, as we found, most (86%) of the commits are posted in regular weekdays. Moreover, the majority (58%) of the commit messages are written in regular working hours while 31% and 11% are found to have been posted respectively in before and after regular working hours. The proportions of commits at different days and periods suggest correctness of the categorization.

In the analysis of the emotional impacts on the lengths of commit messages (Section III-D), we excluded commit messages longer than 50 words, because we observed that commit messages of larger lengths include copy-pasted content such as, SQL statements and code snippets. Such contents are not directly created or typed by the committer and thus are unlikely to reflect his or her emotions.

For the statistical tests of significance in the variations of different distributions, we used the *Mann-Whitney-Wilcoxon (MWW)* test [24]. The *MWW* test is a non-parametric test, which do not require the data to have normal distribution. Since the data in our work do not conform to normal distribution, this particular test suits well for our purpose. Moreover, the significance level α set to 0.05, which is a widely adopted value for this parameter that enables 95% confidence in the results of the *MWW* tests.

External Validity: The findings of this work are based on our study on more than 490 thousand commit messages across 50 open-source projects. This large data-set yields high confidence on the generalizability of the results.

Reliability: The methodology of data collection, analysis, and results are well documented in this paper. The sentiment analysis tool, `SentiStrength` [26] is freely available online and projects studied in this work are also freely accessible through `Boa` [7]. Hence, it should be possible to replicate this study.

V. Related Work

To explore the impacts of emotions on the debugging performance of software developers, Khan et al. [14] used high-arousal-invoking and low-arousal-invoking movie clips to trigger different levels of emotions in developers before having them perform some debugging tasks. However, they did not employ any measurement to extract and quantify the developers emotional states, and relied on the assumption that watching those movie clips would induce different levels of emotions in the developers. Lesiuk [15] recruited 56 software engineers to understand impact of emotions on software design performance. In her work, music was played to arouse developers' positive emotions. The participants self-assessed their emotional states and design performance. Similarly, self-assessment of emotional states were also used in the studies of Wrobel [28] and Graziotin et al. [10].

While the human participants themselves can be expected to accurately report their emotional states, such self-assessment based approaches suffer from the possibility that the participants might be uncomfortable in disclosing their negative emotional states. Biometric measurements such as multi-sensor inputs [18], audio and video processing [29] do not suffer from such difficulties but they can be logistically expensive and difficult for regular use at workplace without disrupting the natural workflow of the practitioners. Both the self-assessment-based and biometric approaches for identification of emotions are difficult (if not impossible) to apply for geographically distributed teams and for extraction of emotions from software artifacts of already completed parts of projects.

Note that, unlike our work, all of the research mentioned above, focused on understanding the *overall* emotional impacts over human performance and indicated positive correlation between them. In contrast, ours include a deeper analysis exploring the impacts and scopes for exploitation of emotions extracted from textual software artifacts such as commit messages. Several other studies also identify developers' emotions from textual software artifacts. In such a study, Murgia et al. [20] reported that issue reports, which express positive emotions take less time to be resolved. They used human raters to identify emotions in issue reports, and thus their work is subject to human errors. Unlike theirs, using an automatic tool `SentiStrength`, we identify emotions in a significantly larger number of commit messages. The automatic tool, `SentiStrength` was also used in the studies of Guzman and Bruegge [13], Tourani et al. [27], Garcia et al. [9], Guzman et al. [12], and in the work of Chowdhury and Hindle [3]. But none of these work tuned the tool before application in software engineering context, as we did in our work.

Guzman and Bruegge [13] identified emotions in collaboration artifacts to relate them with different development topics. In a separate study, using `SentiStrength`, Guzman et al. [12] extracted emotions expressed in 60,425 commit messages and reported that commit comments written on Mondays tend to have more negative emotions compared to Sunday, Tuesday, and Wednesday. However, from the investigation of the same phenomenon using a substantially larger dataset of 490,659 commit messages, our study does not identify any statistically significant variations of emotions in commit comments posted in different days of a week.

Using a Natural Language Toolkit (`NLTK`), Pletea et al. [23] mined developers' emotions from 60,658 commits and 54,892 pull requests for `GitHub` projects. They analyzed emotional variations in discussions on different topics and reported to have found higher negative emotions in security-related discussions in comparison with other topics. While their objective, approach as well as source of emotional content and method of emotion extraction were different from our work, ours includes a deeper and larger analysis based on a larger number of commit messages and diverse aspects of emotional implications.

Using `SentiStrength`, Tourani et al. [27] extracted emotions from emails of both developers and system users. They observed the differences of emotional expressions between developers and users of a system. Using the same tool, Garcia et al. [9] extracted developers' emotions from their email contents to analyze any relationships between

developers' emotions and their activities in an open source software projects. Although the studies of Tourani et al. [27] and Garcia et al. [9] also used the same sentiment analysis tool we used, the source of their emotional content are different and the objectives of those work are also orthogonal to ours.

VI. Conclusion

In this paper, we have presented a quantitative empirical study on the characteristics and impacts of emotions extracted from developers' commit messages. We have studied more than 490 thousand commit comments over 50 open-source projects. Although the majority (65%) of the commit messages are found to be neutral in emotion, surprisingly, positive emotions are found in relatively much smaller portion (13%) of the commit comments than the commits (22%) containing negative emotions.

In our study, we found that the polarities of the developers' emotions significantly vary depending on the type of tasks they are engaged in. The developers express equally high positive and negative emotions in committing in energy-aware tasks compared to other tasks. With respect to the polarities of commit messages, positive emotions are found to be significantly higher than negative emotions in commits for *bug-fixing* and *refactoring* tasks. Surprisingly, the opposite scenario is found for *new feature implementation* tasks.

We also found significant positive correlation between the lengths of commit messages and the emotions expressed in them. When the developers remain emotionally active, they tend to write longer commit comments. However, we did not find any *significant* variations in the developers' emotions in commit messages posted in different times and days of a week.

Based on emotional contents in commit messages, we have also been able to distinguish a group of developers who express more positive emotions at bug-fixing commit messages, another group with the opposite trait, and a third group of developers who equally render both positive and negative emotions at bug-fixing activities. Same approach can be applied for other types of tasks to distinguish potential developers for improved tasks assignment.

The findings from this work are validated in the light of statistical significance. Although more experiments can be conducted to verify or confirm the findings, the results from this study significantly advance our understanding of the impacts of emotions in software development activities and artifacts, and we exemplify how emotional awareness can be exploited in improving software engineering activities.

For automatic computation of emotional polarities in commit messages, we have used a state-of-the-art tool, `SentiStrength`, while alternatives exist. Moreover, before applying the tool, we tuned it for our work in the context of software engineering. In future, we plan to replicate this study using other tools and subjects to further validate the findings of this study. We also have plan to conduct more studies on the impacts of emotions extracted from diverse artifacts including program comments, development forums and email groups.

Acknowledgement: Thanks to Rahul Chatterjee for helping in the statistical analyses.

References

[1] Y. Ayalew and K. Mguniin. An assessment of changeability of open source software. *Computer and Information Science*, 6(3):68–79, 2013.
[2] M. Choudhury and S. Counts. Understanding affect in the workplace via social media. In *CSCW*, pages 303–316, 2013.
[3] S. Chowdhury and A. Hindle. Characterizing energy-aware software projects: Are they different? In *MSR*, pages 1–4, 2016 (to appear).
[4] P. Denning. Moods. *Communications of the ACM*, 55(12):33–35, 2012.
[5] G. Destefanis, M. Ortu, S. Counsell, M. Marchesi, and R. Tonelli. Software development: do good manners matter? *PeerJ PrePrints*, pages 1–17, 2015.
[6] P. Dewan. Towards emotion-based collaborative software engineering. In *CHASE*, pages 109–112, 2015.
[7] R. Dyer, H. Nguyen, H. Rajan, and T. Nguyen. Boa: A language and infrastructure for analyzing ultra-large-scale software repositories. In *ICSE*, pages 422–431, May 2013.
[8] R. Feldt, L. Angelis, R. Torkara, and M. Samuelssonc. Links between the personalities, views and attitudes of software engineers. *Information and Software Technology*, 52(6):611–624, 2010.
[9] D. Garcia, M. Zanetti, and F. Schweitzer. The role of emotions in contributors activity: A case study on the gentoo community. In *CCGC*, pages 410–417, 2013.
[10] D. Graziotin, X. Wang, and P. Abrahamsson. Are happy developers more productive? the correlation of affective states of software developers and their self-assessed productivity. In *PROFES*, pages 50–64, 2013.
[11] D. Graziotin, X. Wang, and P. Abrahamsson. Do feelings matter? on the correlation of affects and the self-assessed productivity in software engineering. *J. of Softw.: Evolution and Proc.*, 27(7):467–487, 2015.
[12] E. Guzman, D. Azócar, and Y. Li. Sentiment analysis of commit comments in github: An empirical study. In *MSR*, pages 352–355, 2014.
[13] E. Guzman and B. Bruegge. Towards emotional awareness in software development teams. In *ESEC/FSE*, pages 671–674, 2013.
[14] I. Khan, W. Brinkman, and R. Hierons. Do moods affect programmers' debug performance? *Cogn. Technol. Work*, 13(4):245–258, 2010.
[15] T. Lesiuk. The effect of music listening on work performance. *Psychology of Music*, 33(2):173–191, 2005.
[16] W. Maalej and H. Happel. From work to word: How do software developers describe their work? In *MSR*, pages 121–130, 2009.
[17] H. Malik, P. Zhao, and M. Godfrey. Going green: An exploratory analysis of energy-related questions. In *MSR*, pages 418–421, 2015.
[18] D. McDuff, A. Karlson, A. Kapoor, A. Roseway, and M. Czerwinski. Affectaura: an intelligent system for emotional memory. In *CHI*, pages 849–858, 2012.
[19] I. Moura, G. Pinto, F. Ebert, and F. Castor. Mining energy-aware commits. In *MSR*, pages 56–67, 2015.
[20] A. Murgia, P. Tourani, B. Adams, and M. Ortu. Do developers feel emotions? an exploratory analysis of emotions in software artifacts. In *MSR*, pages 261–271, 2014.
[21] R. Palacios, A. López, A. Crespo, and P. Acosta. A study of emotions in requirements engineering. *Organizational, Business, and Technological Aspects of the Knowledge Society*, 112:1–7, 2010.
[22] G. Pinto, F. Castor, and Y. Liu. Mining questions about software energy consumption. In *MSR*, pages 22–31, 2014.
[23] D. Pletea, B. Vasilescu, and A. Serebrenik. Security and emotion: Sentiment analysis of security discussions on github. In *MSR*, pages 348–351, 2014.
[24] F. Ramsey and D. Schafer. *The Statistical Sleuth*. Duxbury-Thomson Learning, second edition, 2002.
[25] T. Shaw. The emotions of systems developers: an empirical study of affective events theory. In *SIGMIS CPR*, pages 124–126, 2004.
[26] M. Thelwall, K. Buckley, and G. Paltoglou. Sentiment strength detection for the social web. *Journal of the American Society for Info. Science and Tech.*, 63(1):163–173, 2012.
[27] P. Tourani, Y. Jiang, and B. Adams. Monitoring sentiment in open source mailing lists – exploratory study on the apache ecosystem. In *CASCON*, pages 34–44, 2014.
[28] M. Wrobel. Emotions in the software development process. In *HSI*, pages 518–523, 2013.
[29] Z. Zeng, G. Roisman, and T. Huang. A survey of affect recognition methods: audio, visual, and spontaneous expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(1):39–58, 2009.