Proceedings of the 9th World Congress on Electrical Engineering and Computer Systems and Sciences (EECSS'23)
Brunel University, London, United Kingdom - August 03-05, 2023
Paper No. CIST 112
DOI: 10.11159/cist23.112

Analyzing Emotional Contagion in Merged and Non-Merged Issues in OSS repositories of GitHub

Rashmi Dhakad[1] and Luigi Bendicenti[2]
[1]Faculty of Computer Science
University of New Brunswick
Fredericton NB E3B 5A3, CA
rdhakad@unb.ca; luigi.benedicenti@unb.ca

Abstract - In the past several decades, software engineering has evolved from being confined to in person work and development to being globally distributed. Commit messages based are an important communication aspect of software development platform such as GitHub. A pull request alerts developers that changes have been proposed, review process follows the pull-request that may be merged to the main code. There are many factors that influence the acceptance of these pull-requests. Open-source software is based on collaboration, it is crucial to discover how the emotions expressed by developer discussing affect pull-request issues and influences the acceptance of the pull-request. In this study we analysed the emotional contagion expressed in pull-request issues' whether an issue is merged in the main branch or not.

Keywords**:** Sentiment analysis, emotional contagion, OSS, commits, software development

1. Introduction

A great deal of research has been done on the presence of emotions in software engineering. Understanding the emotions of developers in software development can lead to providing better solutions for developers during the process, which results in a better product at the end. Although sentiment analysis initially was developed as a programmed system for extracting and understanding consumer behaviour and product reviews, in recent times this method has gained popularity with software engineering to solve the issues of collaboration and productivity in software development process.

During the process of software development, a developer goes through several different sentiments throughout the entire process, which have a crucial effect on individual productivity as well as the productivity of the project. Hence if we are able to understand about the sentiments of developer and how these sentiments are being projected to other developers in the team affecting them as well, we will be able to present a better solution.

Previous work done on understanding the sentiments or emotions of developers mainly consisted of analysing the developers' emotions and establishing the relation between individual emotional dynamics with project productivity and collaboration. However, a very limited number of studies have explored the dynamics of contagion of emotions among software developers in the software development.

Emotional contagion by definition can be described as a phenomenon in which the emotional state of one individual has significant effect on the emotional state of the recipient individual. The chain of change in the subsequent emotional states of individuals leads to strong consequential results including the technical factors affecting the project development.

Open-source development has led to breakthrough technologies in the past; thus, understanding the effect of emotional contagion amongst participants in such an environment is important in order to build better tools and provide better services to the developers. Version control systems like GitHub have been proven be effective for open-source software development. Because of its functionality of source code management, pull request feature, ease of integration and access control, it has been used for some of the major open-source technologies in software engineering.

In this study, we perform a qualitative and comprehensive analysis on the 4 open-source projects available on GitHub. We are mining these software repositories in order to gather indirect communication, i.e., sentiments displayed via commit messages, between the developers who contributed towards these open-source projects. Past studies in the field have indicated that the emotional dynamics of developers' sentiments are linked with their productivity and decision-making skills, cognitive abilities, etc. [1], [2]–[4]. Using this as a motivation factor, we focused on commit messages of issues that have been merged in the project and those which are non-merged, detecting contagion in and the consequential resulting factors of the same.

Through this study, we are trying to test the hypotheses presented in our previous study [5]:

- A negative sequence results from a series of non-merged issues stemming from a long period.

- A negative sequence leads to more commits with errors. (We work under the assumption that non- merged issues are the commits with error hence are not merged.)

- A positive sequence leads to a smaller number of commits and issues which results from greater number of merged issues stemmed over the same period.

- We hypothesize that greater number of merged issues in a repository is a result of greater number of positive contagion episodes in comparison to having greater number of negative contagion episodes which are observed in non-merged issues.

We developed an experimental model in which we focused on the merged and the non-merged issues of GitHub. We focused on answering the question of what is the general trend in the developer's commit messages, which have been merged and those which have not been merged. We further delved into the question of finding out whether or not projects with greater number of neutral or positive episodes have more number of merged commits in comparison to projects which have more unresolved issues, i.e. non-merged issues which have greater number of negative sentiment score.

Based on our hypotheses we present the following research question:

RQ1: What is the general sentiment in developer's commit messages which are merged and those which are not?

RQ2: Do projects with a greater number of positive, negative or neutral responses have a significantly greater number of positive, negative and neutral episodes? Do projects with more positive episodes have a smaller number of commits which are merged and do projects with more negative episodes have a greater number of commits which are non-merged?

The rest of the paper is divided into the following sections. Section 2 discusses the related work and other background work done by researchers in the past regarding the emotions of software developers and their consequences, Section 3 describes the sentiment analysis approach followed for the study, followed by Section 4 in which we discuss the findings and results of the analysis. We conclude our study in Section 6 with future work and expectations.Papers should clearly describe the background of the subject, the authors work, including the methods used, results and concluding discussion on the importance of the work. Papers are to be prepared in English and SI units must be used. Technical terms should be explained unless they may be considered to be known to the conference community.

2. Related Work

For this research study, we intend to cite examples of literature regarding the link between software development process and developer's sentiments which encouraged our findings and our work. Previous studies in the domain of software engineering have been done on the sentiment analysis of software developers. Graziotin et al. in their work [6]–[9], have analyzed the sentiments of software developers and contributed to the understanding of their effects on the development process. In our previous work, we concentrated our efforts on establishing the relationship between developer's emotional state amongst each other. Our work primarily focused on determining the presence of emotional contagion amongst software developers in an open source development process. For this study as well as our current study we benefitted from the open source repositories of GitHub along with the archived data of GH Torrent[10].

Previous literature has presented evidence that suggests that the negative emotions often result in consequences which impact the overall software development process[9], [11] ,[12]. In the study [12], Destefanis et. Al have mentioned that positive parallel amongst developers leads to better relationships and reduced time for bug fixing. Similar to this, Graziotin et al. [9] conducted a qualitative analysis of more than 150 participants who are in active state of development process. They found out that unhappiness which can be broadly termed as a negative sentiment or mood has contributed towards developer's taking shortcuts and poor quality of work and productivity performance.

Another study conducted by authors Ortu et al. focused on establishing the relationship between developer's sentiment and the time to fix issues in over 560k issue comments in Jira. The authors observed that positive emotions are correlated with less time to resolve the issues, whereas negative emotions like sadness were associated with longer fixing times.

Although we found previous literature related to developer's sentiment, and other researchers have done studies that that compile the analysis and studies done on commit messages and merged and non-merged issues, and the sentimental analysis and its consequences for the overall software development process or the developer itself, we did not find any literature which strongly focused on the emotional contagion aspect of merged and non-merged issues, which is what we try to focus on in this study.

3. Experimental Setup/Dataset Used

The study is designed to investigate how affect or emotions expressed in the merged commit messages have an impact on the overall issue resolution. For the successful analysis of the research questions, we opted for sample datasets on which we have performed previous experiments in detecting emotional contagion, keeping in mind that the previous experiment consisted of mining the commit messages from all participants, irrespective of their merged status, and these were used for further analysis and insight. For the purpose of relevancy, we used the same datasets which included of a large number of participants. In order to achieve this, we made use of GitHub's open-source repositories.

GitHub is an online version control system which provides the users different functionalities to collaborate and track their work on projects from anywhere in the world.

For the purpose of our analysis, we extracted our data from GitHub's open-source repositories; this dataset included four open-source project repositories including VsCode, Tensorflow, Pandas, PyTorch. In order to mine the required information, we first cloned each project repository onto the local machine (x64-based processor, 1.80GHz 1.99 GHz, 16 GB RAM), using Git commands.

```
rashm@Rashmi-1 MINGW64 ~/tensorflow (master)
$ git log --no-merges --date=format:'%Y-%m-%d %H:%M:%S' --pretty=format:'"%h","%an","%ad","%s","%b"'>tensorflow_no_merges.csv
```

Figure 1: Fetching non-merged contents for dataset from the commit logs of Tensorflow

After the successful mining of desired data, we formatted our dataset into comma-separated files which can be easily processed by our analytical tool.

4. Sentiment Analysis Approach

To determine the sentiment polarity expressed in the merged or non-merged issue commit messages by developers' we used Wolfram Mathematica. Mathematica is a general-purpose computational programming tool and has several built-in functionalities for complex computational problems like computer vision, image processing, data science, neural networks, machine learning, etc., which provides the analytical functionalities that were required for this analysis. In comparison to other tools used for sentiment analysis, this tool provides a wider range of functionalities for performing multiple analysis and visualizations for results in a wide range of industry from biology to geo-socio science.

Mathematica has been used in our previous study [5], where we established the presence of emotional contagion in open source commit messages, our goal in the previous study of chapter 4 was to demonstrate that emotional contagion exists and to identify the episodes of contagion using a trained classifier which is available in Mathematica.

Similar to our previous study of chapter 4, in this study we first establish the presence of emotional contagion in commit messages which are categorized into merged and non-merged issues, by investigating various episodes of negative and positive, we further analyze the affect episodes in both the categories and try to hypothesize the connection between merged commit messages and positive episodes of contagion and negative episodes and non- merged commit messages .

Firstly, we gather all the commit messages from the open-source repository; after cleaning up the undesired information from the dataset, we categorize the data into merged issue commit messages and non-merged issues commit messages. Further, we perform a sentiment polarity analysis on each category of data.

Using Mathematica's built-in classifier, Classify [16], we first analyze the commit messages' polarity of positive, negative and neutral. The resulting analysis is then converted into a numerical format of -1, 0,+1, where -1,0,1 indicates commit message with negative, neutral and positive polarity, respectively. The ensuing numerical sequence is then against time each commit message is made and plotted.

In order to keep the scope of the study under control, we adopted some predefined constraints. The first assumption is with regard to the dataset, we assume that the contributors in the projects of all the repositories are geographically located and do not have any in-person or face-to-face communication.

Furthermore, we base our experiment on the conjecture that participants of all the merged issues are significantly affected by the emotional dynamics presented to them during the process. We also worked under the assumption that from the time the participant was exposed to the emotional dynamics of the sender, there was no other factor that initiated any change in the emotional dynamics of the recipient.

With the understanding of the presence of contagion in merged issues and non-merged issues, we will be able to further answer questions which are based on the consequential results of contagion.

To illustrate the working of the experiment, we will provide a simplified example to illustrate its procedures, keeping in mind that this example is more simplified rather than being widespread as to keep it brief.

In order to acquire the dataset, we used the following Git log commands which provided us with a dataset of all the commit messages which have been merged in a repository.

The desired dataset was stored in a comma separated file so as to easily be readable by the analytics tool. Here our desired tool for analysis is Wolfram Mathematica[17].

After the successful retrieval of desired dataset, we imported the dataset into Mathematica and proceeded to remove any anomalies in the dataset.

```
mergepandas = Dataset[Import["C:\\Users\\rashm\\Dropbox\\Research\\My work\\New Paper- January 2022\\Database\\Pandas\\pandas_merge.csv", "CSV", "Numeric" → False]]
```

81dfb288e7	Jeff Reback	2016-01-21 10:37:15	Merge pull request #12111 from jreback/matrix
ee9fbb6c78	Jeff Reback	2016-01-21 07:26:50	Merge pull request #12109 from kawochen/ENH-12034-union
a783d1d4f8	Tom Augspurger	2016-01-20 17:55:53	Merge pull request #12090 from HHammond/master
031d7caa34	Jeff Reback	2016-01-17 11:21:48	Merge pull request #12068 from jreback/wheel2
270dac623a	Jeff Reback	2016-01-16 22:20:24	Merge pull request #12065 from jreback/wheel
723a147a4a	Jeff Reback	2016-01-16 12:36:53	Merge pull request #11892 from jreback/ri
efb2e906c2	Jeff Reback	2016-01-16 12:33:32	Merge pull request #12059 from rishipuri/master
ce37586660	Jeff Reback	2016-01-16 11:30:52	Merge pull request #12058 from jreback/array_equivalent

Figure 2: Importing the dataset into Mathematica for Merged Commit Messages of Pandas

```
mergepandas = mergepandas[Select[#Message ≠ "" &], All]
```

Commit	Author	Date	Subject
81dfb288e7	Jeff Reback	2016-01-21 10:37:15	Merge pull request #12111 from jreback/matrix
ee9fbb6c78	Jeff Reback	2016-01-21 07:26:50	Merge pull request #12109 from kawochen/ENH-12034-union
a783d1d4f8	Tom Augspurger	2016-01-20 17:55:53	Merge pull request #12090 from HHammond/master
031d7caa34	Jeff Reback	2016-01-17 11:21:48	Merge pull request #12068 from jreback/wheel2
270dac623a	Jeff Reback	2016-01-16 22:20:24	Merge pull request #12065 from jreback/wheel
723a147a4a	Jeff Reback	2016-01-16 12:36:53	Merge pull request #11892 from jreback/ri

Figure 3: Removing anomalous data from the dataset of Merged Commit Messages of Pandas

Afterwards, we proceeded to format the data according to the timestamp of each commit comment, which gave us a formatted dataset with time and complete commit messages.

```
mergepandas = MapAt[DateObject, Rest@mergepandas, {All, 3}]
```

Commit	Author	Date	Subject
ee9fbb6c78	Jeff Reback	Thu 21 Jan 2016 0	Merge pull request #12109 from kawochen/ENH-12034-union
a783d1d4f8	Tom Augspurger	Wed 20 Jan 2016 1	Merge pull request #12090 from HHammond/master
031d7caa34	Jeff Reback	Sun 17 Jan 2016 1	Merge pull request #12068 from jreback/wheel2
270dac623a	Jeff Reback	Sat 16 Jan 2016 22	Merge pull request #12065 from jreback/wheel
723a147a4a	Jeff Reback	Sat 16 Jan 2016 12	Merge pull request #11892 from jreback/ri
efb2e906c2	Jeff Reback	Sat 16 Jan 2016 12	Merge pull request #12059 from rishipuri/master

Figure 4: Formatted Dataset for Merged Commit Messages of Pandas

The formatted dataset is then used for further sentiment analysis. All the commit messages are classified using Mathematica's built-in classifier function, Classify. The function classifies the sentiment of a text into 4 given classes, i.e. positive, negative, neutral and indeterminate. The indeterminate value indicate that Mathematica could not classify the sentiment polarity of the text.

```
BarChart[Apply[Labeled, Reverse[{{"Negative", 1250}, {"Positive", 480}, {"Neutral", 1294}, {Indeterminate, 8}}, 2], {1}]]
```

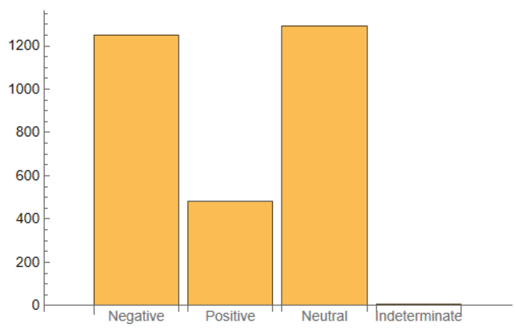

Figure 5: Sentiment Polarity of Merged Commit Messages in Pandas.

As we can observe, a vast majority of the cases are neutral which is closely followed by negative polarity, which indicates towards different prognosis than what we initially expected. Since a very small number of cases are classified as indeterminate, in order to keep the relevancy of the experiment results, we considered these cases as neutral. After the successful polarity analysis of the sentiments of the messages, we then converted the resulting classes into numerical values of 1, 0, -1 for positive, neutral and negative respectively.

```
{<|Commit → 52f5b454a6, Author → Wes McKinney, Date → Tue 28 Jun 2011 19:06:47 GMT-3,
 Subject → Merge pull request =55 from dieterv77/CleanCommand, Message → Minor change to CleanCommand so build works with stdeb, Sentiment → -1|>,
 <|Commit → 64ea7473f6, Author → Wes McKinney, Date → Tue 28 Jun 2011 22:07:34 GMT-3, Subject → Merge branch 'master' of github.com:wesm/pandas, Message → * 'master' of github.com:wesm/pandas:
 Minor change to CleanCommand so build works with stdeb
, Sentiment → 1|>, <|Commit → 7e634dec1b, Author → Wes McKinney, Date → Fri 1 Jul 2011 09:10:01 GMT-3, Subject → Merge branch 'master' into widepanel-refactor, Message → * master:
 testing IO roundtripping with HDFStore
 added ascending argument to Series.order
, Sentiment → 0|>, <|Commit → 46b58d0d6b, Author → Wes McKinney, Date → Fri 1 Jul 2011 14:31:53 GMT-3, Subject → Merge branch 'master' into widepanel-refactor, Message → * master:
 BUG: fixed join_on regression from refactor
, Sentiment → 1|>, <|Commit → 7cef2f1597, Author → Wes McKinney, Date → Fri 1 Jul 2011 17:15:31 GMT-3, Subject → Merge branch 'master' into widepanel-refactor, Message → * master:
 BUG: support legacy legacy DataFrame pickle
, Sentiment → 1|>}
```

Figure 6: Sentiment polarity classification of merged commit messages.

After the successful analysis of the polarity of each text, we then mapped the polarity of the commit messages to the respective timeline, which gave us a graph of overall sentiment polarity of the repository for the given timeframe of the dataset.

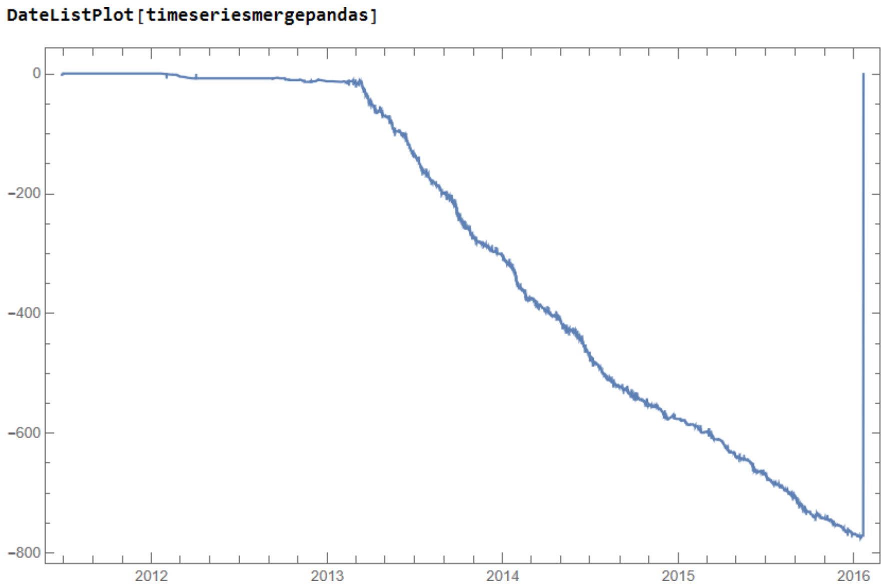

Figure 7: Sentiment polarity of merged commit messages of Pandas

The above graph in the figure 7 depicts the contagious affect of positive, negative and neutral responses. In order to understand the contagious affect in merged and non-merged issues, we will exemplify a small portion of this data to further explain and theorize on our hypothesis and research questions.

After the successful completion of identifying the contagious affects of the issues we further delved into the dissecting our results of contagion by length of the effect. For this purpose, we carried out further analysis which involved further dissecting the data to determine the length of contagion cases.

For example, in the case of merged issues in the Pandas repository, we found that there are a total of 12 cases which can be considered a positive contagious effect as the length of the contagion is greater than 3 responses at a time. This means that after the first positive response there are 2 more consecutive responses that happened which changed the trajectory of contagion.

Here we are working under the assumption that episodes which have only two consecutive similar responses are not cases of contagion; this is because in order for a contagion to be considered an episode it has to have a continuous set of repetitive response polarity, i.e. the sentiment polarity of the text should be same as the predecessor and the successor should follow suit.

```
In[23]:= tempmp4positive = SequenceCases[tempmp4, {1, 1 ..}] // Tally // Grid

Out[23]=       {1, 1}        56
             {1, 1, 1}        9
            {1, 1, 1, 1}      2
         {1, 1, 1, 1, 1, 1, 1} 1

In[24]:= tempmp4neutral = SequenceCases[tempmp4, {0, 0 ..}] // Tally // Grid

Out[24]=   {0, 0, 0, 0, 0, 0, 0, 0, 0}       1
                  {0, 0}                171
         {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}  1
                {0, 0, 0, 0}             31
                 {0, 0, 0}               86
               {0, 0, 0, 0, 0}           13
              {0, 0, 0, 0, 0, 0}          8
             {0, 0, 0, 0, 0, 0, 0}        1

In[25]:= tempmp4negative = SequenceCases[tempmp4, {-1, -1 ..}] // Tally // Grid

Out[25]=          {-1, -1}              179
           {-1, -1, -1, -1, -1}         13
             {-1, -1, -1, -1}           30
         {-1, -1, -1, -1, -1, -1, -1}   4
               {-1, -1, -1}             73
          {-1, -1, -1, -1, -1, -1}      2
```

Figure 8: Different contagion episodes in Merged issue commits in Pandas repository

Based on the above observations and analytical results we then try to answer our initial hypothesis and research questions in the next section, Results and Discussion.

5. Results

To investigate the research questions, we created a timeline graphs that depict the emotional contagion effects in merged and non-merged issues and sentiment score of commit messages of these issues. We began with analyzing the entire dataset for each repository at once.

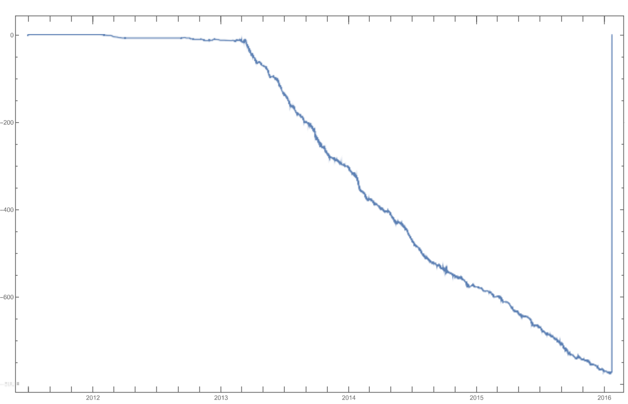

Figure 9(a): Cumulative emotional contagion plot for all merged issues for Pandas

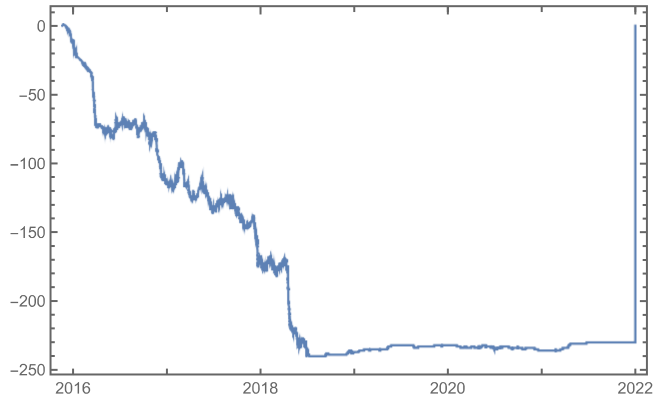

Figure 9(b): Cumulative emotional contagion plot for all merged issues for Tensorflow

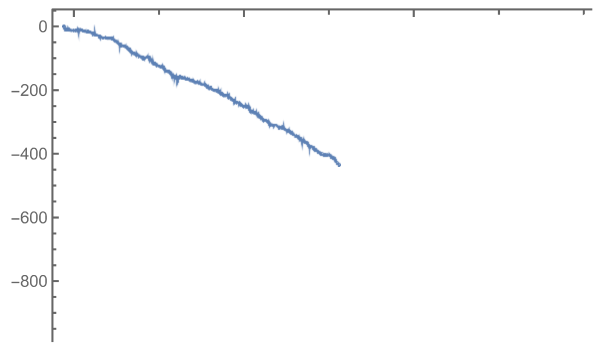

Figure 9(c): Cumulative emotional contagion plot for all merged issues for VsCode

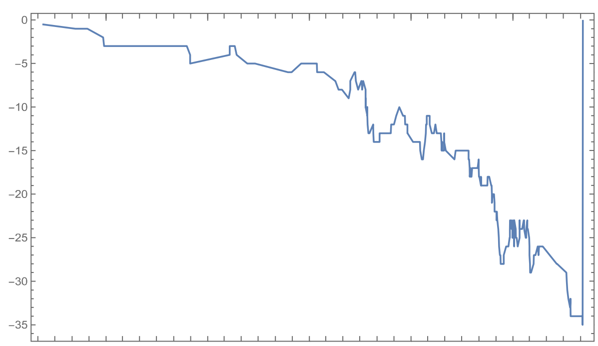

Figure 9(d): Cumulative emotional contagion plot for all merged issues for PyTorch

Here for repositories Pandas and Pytorch, upon retrieval and cloning the repository, all the merged issues for these two repositories have the latest by 2016 and 2017 respectively, whereas for the remaining repositories Tensorflow and VsCode, there have been more data available till 2022. Here we are working under the assumption that all the merged issue commit messages have been posted and date correctly and that there is no irregularity in the dataset. The categorization amongst the different sentiment polarity includes positive, negative, neutral and indeterminate. We have considered the cases which have been labeled indeterminate by Mathematica as neutral cases and have thus assigned them 0.

RQ1: What is the general sentiment in developer's commit messages which are merged and those which are non-merged?

We analyzed the commit messages of issues which are merged and issues which are non-merged and analyzed the sentiment polarity index, and compared the results and possible outcomes.

Motivation: In the past few decades there have been a vast number of studies that have investigated the emotions of of software developers, and how these emotions influence the performance of the software developer. Previous studies also also indicated towards how emotions of software developers affect their productivity and performance [7]. In this study we we investigated the contagious affect of positive, negative or neutral sentiments in commit messages which are merged, as as well as those which non-merged.

Findings: The graphs shown below account for the positive, negative, neutral, responses in merged as well as non-merged issues of commit messages from all different repositories' datasets.

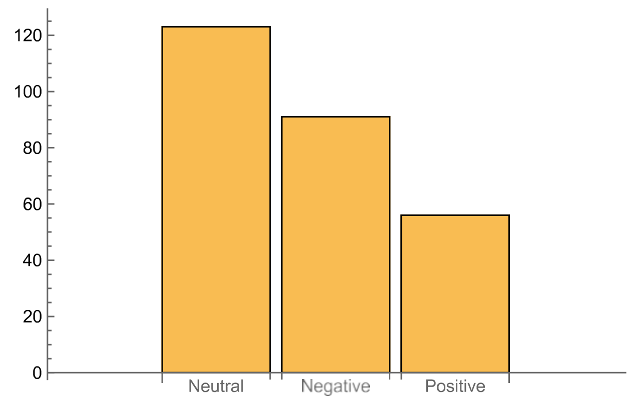

Figure 10(a): Merged issues sentiment score in Pandas

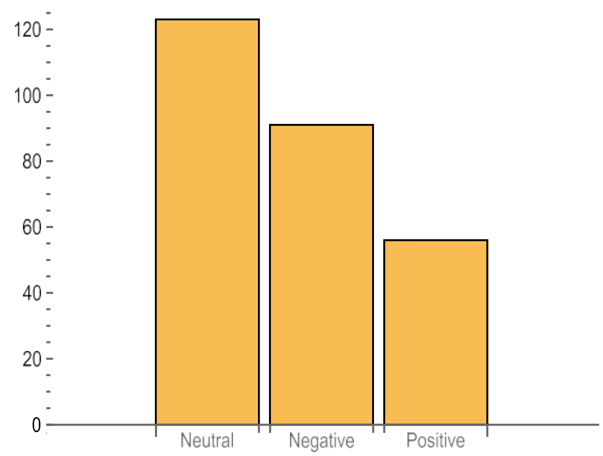

Figure 10(b): Merged issues sentiment score in PyTorch

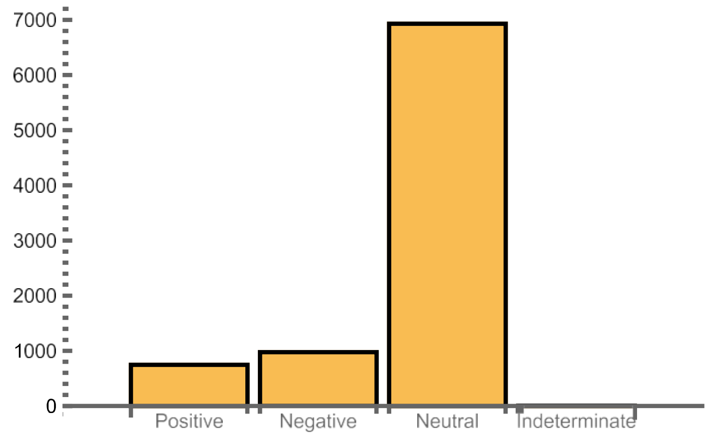

Figure 10(c): Merged issues sentiment score in TensorFlow

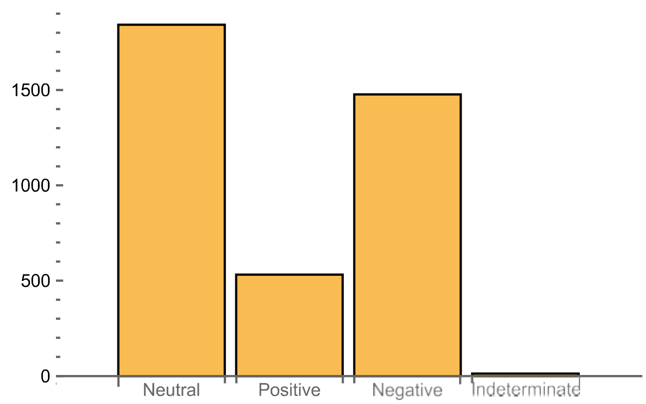

Figure 10(d): Merged issues sentiment score in VsCode

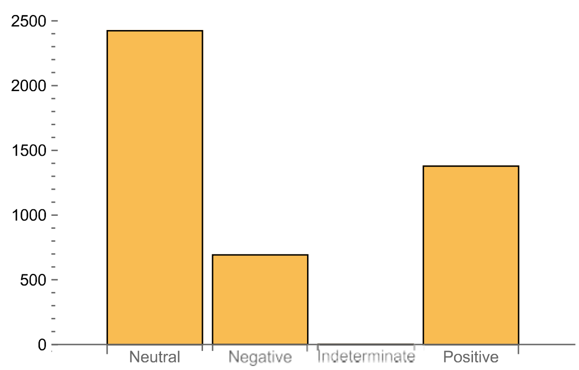

Figure 11(a): Non-Merged issues sentiment score in Pandas

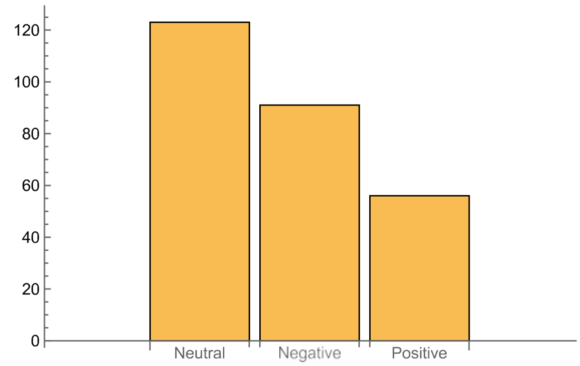

Figure 11(b): Non-Merged issues sentiment score in PyTorch

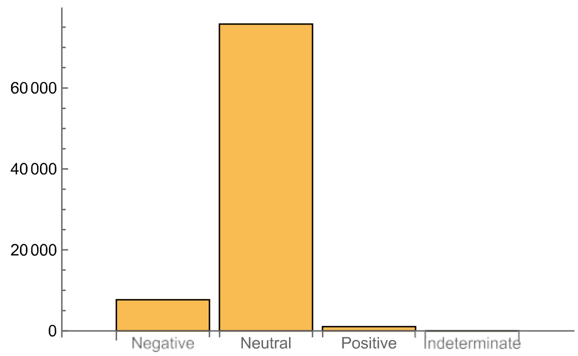

Figure 11(c): Non-Merged issues sentiment score in TensorFlow

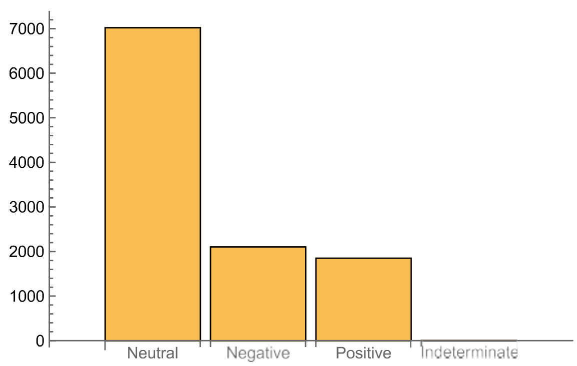

Figure 11(d): Non-Merged issues sentiment score in VsCode

From the statistics, for merged issues, in case of the repository Pandas, there are fewer positive messages in comparison to negative, along with this we find that the number of neutral messages are more than negative, which contradicts to our initial hypotheses presented in our earlier study[5].

Similarly, with other repositories, we find the same contradicting statistics, although, in case of repository Tensorflow, we see a big difference in the total number positive and negative messages to that of neutral messages.

RQ2: Does projects with a greater number of positive, negative or neutral responses have significantly a greater number of positive, negative and neutral episodes? Does projects with more positive episodes have fewer number of commits which are merged and does projects with more negative episodes have greater number of commits which are non-merged?

Motivation: In previous literature studies, researchers have found relationship between emotional state of developer to that of their productivity[2], furthermore different researchers have found evidences on software developer's emotions and their coding practices and performance [18]. Different emotions and sentiments have proved to be affecting the software development process [9] [19]. In this study we try to associate the relationship between emotional contagion episodes and its consequences on software development process.

Findings: In order to investigate whether or not a greater number of different positive, negative, neutral response results in a greater number of positive, negative and neutral episodes, we took small sample size from each repository and observed the resulting sequence of emotional contagion affects. In the following graphs we can see that for the project repository Pandas, for a sample datapoints of 201 datapoints, total number of positive responses is 33, for negative is 85 and 83 for neutral. Whilst the negative responses are greater in number, the total number of neutral emotional contagion episodes are more than that of negative emotional contagion with positive emotional contagion coming last in position.

Furthermore, in case of the project repository Pytorch, the total number of positive, negative, neutral response is relative to that of the total number of emotional contagion sequences of positive, negative, and neutral contagion episodes. A similar trend has been observed in VsCode, where the total number of neutral contagion episodes recorded were 263 with total number of neutral responses being 1855, and number of negative contagion episodes being 135, against 1477 negative responses. With project repository Tensorflow, we observed that the total number of neutral contagion episodes are 129, against 6937 neutral responses, and total 88 negative emotional contagion episodes were recorded against the 975 negative responses.

```
In[84]:= tempmp4positive = SequenceCases[tempmp4, {1, 1 ..}] // Tally // Grid

Out[84]=        {1, 1}              56
              {1, 1, 1}              9
            {1, 1, 1, 1}             2
        {1, 1, 1, 1, 1, 1, 1}        1

In[85]:= tempmp4neutral = SequenceCases[tempmp4, {0, 0 ..}] // Tally // Grid

Out[85]=   {0, 0, 0, 0, 0, 0, 0, 0, 0}         1
                   {0, 0}                    171
        {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}      1
                {0, 0, 0, 0}                  31
                 {0, 0, 0}                    86
              {0, 0, 0, 0, 0}                 13
            {0, 0, 0, 0, 0, 0}                 8
          {0, 0, 0, 0, 0, 0, 0}                1

In[86]:= tempmp4negative = SequenceCases[tempmp4, {-1, -1 ..}] // Tally // Grid

Out[86]=          {-1, -1}                   179
          {-1, -1, -1, -1, -1}                13
            {-1, -1, -1, -1}                  30
      {-1, -1, -1, -1, -1, -1, -1}             4
              {-1, -1, -1}                    73
         {-1, -1, -1, -1, -1, -1}              2
```

Figure 12: Total number of different sentiment score sequences in merged issues of Pandas for 201 datapoints

```
In[]:= tempmt4negative = SequenceCases[tempmt4, {-1, -1, -1 ..}] // Tally // Grid

Out[]=         {-1, -1, -1, -1, -1}                 10
                  {-1, -1, -1}                      55
          {-1, -1, -1, -1, -1, -1, -1}               1
                {-1, -1, -1, -1}                    19
            {-1, -1, -1, -1, -1, -1}                 2
    {-1, -1, -1, -1, -1, -1, -1, -1, -1, -1}         1
```

Figure 13(a): Total number of negative sentiment score sequences in merged issues of Tensorflow

```
In[145]:= t1positive = SequenceCases[t1, {1, 1 ..}] // Tally // Grid

Out[145]=    {1, 1}    10
          {1, 1, 1}    3

In[146]:= t1negative = SequenceCases[t1, {-1, -1 ..}] // Tally // Grid

                  {-1, -1}           19
              {-1, -1, -1}            4
Out[146]= {-1, -1, -1, -1, -1}        1
            {-1, -1, -1, -1}          1

In[147]:= t1neutral = SequenceCases[t1, {0, 0 ..}] // Tally // Grid

                   {0, 0}           15
                {0, 0, 0}            9
Out[147]= {0, 0, 0, 0, 0, 0}         3
             {0, 0, 0, 0, 0}         1
               {0, 0, 0, 0}          2
```

Figure 13(b): Total number of different sentiment score sequences in merged issues of Pytorch

```
In[170]:= tempmv4positive = SequenceCases[tempmv4, {1, 1 ..}] // Tally // Grid

                    {1, 1}                74
Out[170]=        {1, 1, 1}                 7
          {1, 1, 1, 1, 1, 1, 1, 1, 1}      1

In[171]:= tempmv4negative = SequenceCases[tempmv4, {-1, -1 ..}] // Tally // Grid

                       {-1, -1, -1}                          77
                         {-1, -1}                           216
                  {-1, -1, -1, -1, -1}                       12
                      {-1, -1, -1, -1}                       34
Out[=]=         {-1, -1, -1, -1, -1, -1, -1}                  3
                   {-1, -1, -1, -1, -1, -1}                   6
              {-1, -1, -1, -1, -1, -1, -1, -1}                2
          {-1, -1, -1, -1, -1, -1, -1, -1, -1, -1}            1

In[=]:= tempmv4neutral = SequenceCases[tempmv4, {0, 0 ..}] // Tally // Grid

                    {0, 0}                   226
                  {0, 0, 0}                  114
               {0, 0, 0, 0, 0}                34
                 {0, 0, 0, 0}                 53
Out[=]=    {0, 0, 0, 0, 0, 0, 0, 0, 0}         4
            {0, 0, 0, 0, 0, 0, 0, 0}           7
             {0, 0, 0, 0, 0, 0}               10
            {0, 0, 0, 0, 0, 0, 0}              4
         {0, 0, 0, 0, 0, 0, 0, 0, 0, 0}        1
```

Figure 13(c): Total number of different sentiment score sequences in merged issues of VsCode

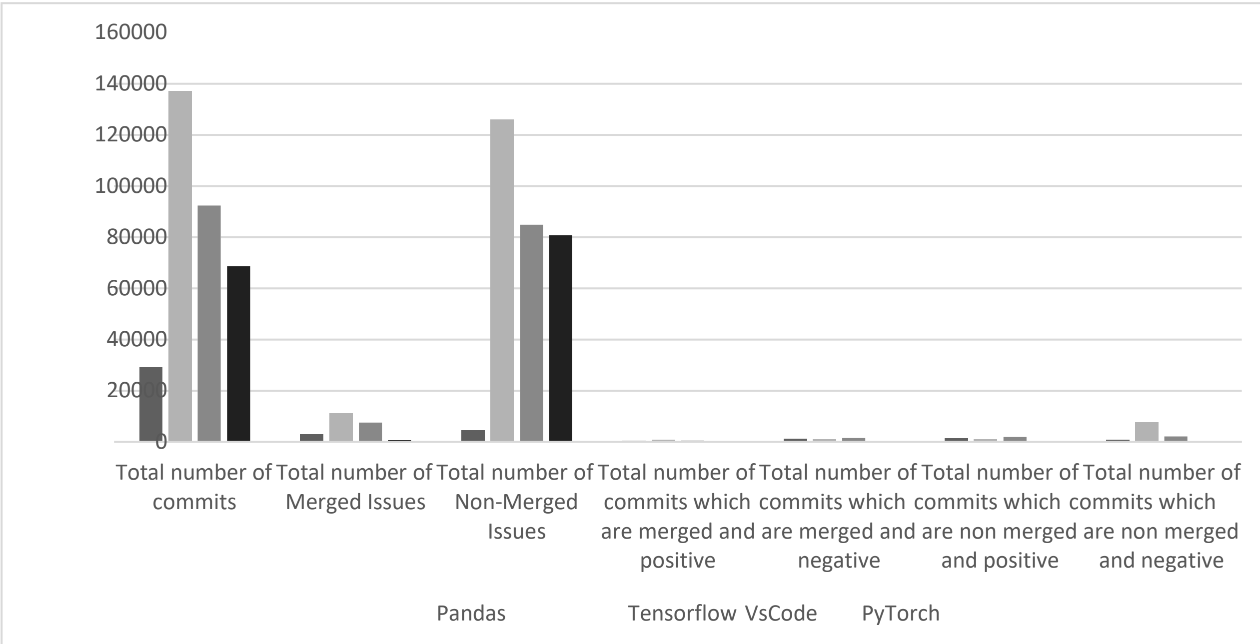

Figure 14: Number of positive and negative merged and non-merged commits

In case of merged issues of the project Pandas, the total number of commits are 29271, whilst the total number of merged issues are 3032, and the total number of commits which are merged and positive are 480, in comparison to the total number of non-merged issues which are positive 1378. The total number of negative responses which are merged are 1250 whereas total number of non-merged issues which are negative are 692. This negates our initial hypothesis that the merged issues are a result of a greater number of positive responses, where are non-merged issues are a result of more number of negative issues. Additionally, in case of the project repository Tensorflow the total number of commits are 137210, while the total number of merged and positive commits are 746, and the total number of positive and non-merged issues are 1049. Moreover, the total number of issues which are negative and merged are 975, while total number of issues which are negative and non-merged are 7669. We observe a similar case with project repositories VsCode. Whereas in case of project repository PyTorch, the total number of positive and negative responses for both type of issues is the same. What we observe is that despite being merged, there are a greater number of negative commits and fewer number of positive commits. Whereas, in non-merged issues, there are a greater number of positive commits and lesser number of negative commits.

6. Conclusions and Future Work

Throughout the data we have analyzed, we find the evidence that issues which have been merged have no correlation with positive contagion episodes, i.e. having more number of positive responses or more number of positive contagion episodes does not necessarily mean that the issues are going to be merged. Furthermore, non- merged issues doesn't have more number of negative emotional contagion episodes which indicates that issues which are not merged doesn't mean that they are a result of periodically stemmed negative emotions.

Further research can be done on this topic to analyze the relationship between the consequential effect of positive or negative emotional contagion on issue reopening or merging and non- merging of issues. We commend a greater scale study which includes a greater number of authors and a greater number of issues in order to compare the relationship between merged issues and different emotional contagion episodes. Furthermore, we recommend analyzing the relationship between different type of emotional contagion, for example, happiness, sadness, anger, etc. on open-source software projects in order to analyze how different emotions affect the overall development process differently.

References

[1] D. Graziotin, X. Wang, and P. Abrahamsson, "Do feelings matter? On the correlation of affects and the self- assessed productivity in software engineering," J. Softw. Evol. Process, vol. 27, no. 7, pp. 467–487, 2015, doi: 10.1002/smr.1673.

[2] M. Anany, H. Hussien, S. Aly, and N. Sakr, "Influence of Emotions on Software Developer Productivity," Jun. 2020, pp. 75–82. Accessed: Jun. 25, 2020. [Online]. Available: https://www.scitepress.org/Link.aspx?doi=10.5220/0008068800750082

[3] B. Crawford, R. Soto, C. L. de la Barra, K. Crawford, and E. Olguín, "The Influence of Emotions on Productivity in Software Engineering," in HCI International 2014 - Posters' Extended Abstracts, Cham, 2014, pp. 307–310. doi: 10.1007/978-3-319-07857-1_54.

[4] H. R. Heekeren, S. Schulreich, P. N. C. Mohr, and C. Morawetz, "How incidental affect and emotion regulation modulate decision making under risk," in 2017 5th International Winter Conference on Brain- Computer Interface (BCI), Jan. 2017, pp. 13–15. doi: 10.1109/IWW-BCI.2017.7858145.

[5] R. Dhakad and L. Benedicenti, "Detecting Emotional Contagion in OSS Projects," presented at the The 7th World Congress on Electrical Engineering and Computer Systems and Science, Aug. 2021. doi: 10.11159/cist21.301.

[6] D. Graziotin, X. Wang, and P. Abrahamsson, "Are Happy Developers More Productive?," in Product- Focused Software Process Improvement, Berlin, Heidelberg, 2013, pp. 50–64. doi: 10.1007/978-3-642-39259- 7_7.

[7] D. Graziotin, X. Wang, and P. Abrahamsson, "Software Developers, Moods, Emotions, and Performance," IEEE Softw., vol. 31, no. 4, pp. 24–27, Jul. 2014, doi: 10.1109/MS.2014.94.

[8] D. Graziotin, X. Wang, and P. Abrahamsson, "Understanding the affect of developers: theoretical background and guidelines for psychoempirical software engineering," in Proceedings of the 7th International Workshop on Social Software Engineering, Bergamo, Italy, Sep. 2015, pp. 25–32. doi: 10.1145/2804381.2804386.

[9] D. Graziotin, F. Fagerholm, X. Wang, and P. Abrahamsson, "Unhappy Developers: Bad for Themselves, Bad for Process, and Bad for Software Product," in 2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C), May 2017, pp. 362–364. doi: 10.1109/ICSE-C.2017.104.

[10] "The GHTorrent project." https://ghtorrent.org/ (accessed Jan. 25, 2022).

[11] S. F. Huq, A. Z. Sadiq, and K. Sakib, "Is Developer Sentiment Related to Software Bugs: An Exploratory Study on GitHub Commits," in 2020 IEEE 27th International Conference on Software Analysis, Evolution and Reengineering (SANER), Feb. 2020, pp. 527–531. doi: 10.1109/SANER48275.2020.9054801.

[12] G. Destefanis, M. Ortu, S. Counsell, S. Swift, M. Marchesi, and R. Tonelli, "Software development: do good manners matter?," PeerJ Comput. Sci., vol. 2, p. e73, Jul. 2016, doi: 10.7717/peerj-cs.73.

[13] V. Sinha, A. Lazar, and B. Sharif, "Analyzing developer sentiment in commit logs," in Proceedings of the 13th International Conference on Mining Software Repositories, New York, NY, USA, May 2016, pp. 520– 523. doi: 10.1145/2901739.2903501.

[14] 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, Hyderabad, India, May 2014, pp. 352–355. doi: 10.1145/2597073.2597118.

[15] E. Guzman, "Visualizing emotions in software development projects," in 2013 First IEEE Working Conference on Software Visualization (VISSOFT), Sep. 2013, pp. 1–4. doi: 10.1109/VISSOFT.2013.6650529.

[16] "Classify—Wolfram Language Documentation." https://reference.wolfram.com/language/ref/Classify.html (accessed Jul. 13, 2021).

[17] "Wolfram Mathematica: Modern Technical Computing." https://www.wolfram.com/mathematica/ (accessed Jul. 13, 2021).

[18] M. R. Anany, H. M. W. Hussein, and S. G. Aly, "A survey on the influence of developer emotions on software coding productivity," Int. J. Soc. Humanist. Comput., vol. 3, no. 3/4, p. 216, 2020, doi: 10.1504/IJSHC.2020.111166.

[19] S. C. Müller and T. Fritz, “Stuck and Frustrated or in Flow and Happy: Sensing Developers’ Emotions and Progress,” in 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering, May 2015, vol. 1, pp. 688–699. doi: 10.1109/ICSE.2015.334.