CS 5306: Project 1 Report

Analysis of Github Contribution in Connection to Project Sentiment

Natasha Armbrust (nka8)   Fatima AlGhamdi (faa52)

Introduction

In this paper, we analyze the crowdsourcing platform Github, a web-based version control repository that allows easy project collaboration and is widely used for open-source software projects. As computer scientists, software collaboration is an important and crucial part of software development. We are particularly curious in determining factors behind the amount of contributions to a project and how these factors can influence the progress of contributions over time. The factor we are targeting in this paper is project sentiment. We believe tone and sentiment is very important for group collaboration. Thus, we wanted to turn to Github, an online group collaboration setting, to analyze sentiment in a virtual crowdsourcing platform.

This paper's objective is to answer the question of if project sentiment correlates with project contribution. To answer this question we completed sentiment analysis on pull request comments using the dataset MSR 2014 Mining Challenge Dataset [1]. This dataset provides data from the top-10 starred software projects for the top programming languages on Github, which gives 90 projects total. The dataset includes projects, users, pull requests, pull request comments, and pull request history and can be downloaded and accessed through MySQL and MongoDB. We implemented a simple sentiment analysis model to do sentiment analysis on pull request comments to get the percentage of positive, negative, and neutrality in the comment. Project contribution amount was assessed via commits to the repositories. Both total commits and pull request comment sentiment were measured as a function of time. The analysis in this paper is driven by data visualizations.

Data

Our project is developed using the data from MSR 2014 Mining Challenge Dataset provided by Github [1]. The total dataset contains 10,8718 projects, 555,325 commits, and 54,892 pull request comments. However, only 90 of the 10,8718 are the original top repositories on Github (the other projects are forks with the base repo being one of the original top repositories). Our analysis contains two components: contributions over time and pull-request sentiment analysis. We also merge the results into combined visualizations.

Methodology

Contributions Over Time

To determine contributions over time, we looked at commit contributions and the equivalent date they were created. We used MySQL database provided by MSR 2014 Mining Challenge Dataset [1]. We used SQL queries to select project_id, created_at from the commit table for each of the top 90 projects. We used Python as our programming language for aggregating the data and creating visualizations with help from libraries numpy and matlibplot [2][3].

Pull-Request Sentiment Analysis

The data used for sentiment analysis was retrieved from pull_request_comments table. We selected id, body(comment), created_at from the pull_request_comments table and joined it with the projects table to get the data where the base repo was one of the top 90 Github projects. The total number of comments retrieved was around 54,000. Of the total 90 repos we analyzed, on a per repo basis, the maximum number of pull request comment data was 13171 and the minimum number was 2. For preprocessing the data, we used Vadar provided by Natural Language ToolKit [4] [5]. Vadar is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media. This fits Github comments because they are often short and in the phrase and tone of comments on various social media sites (except for the Computer Science lingo). Given a comment, Vadar returns the percentage negative, positive, and neutrality of the comment. In our analysis we used the compounded score which takes into account negative, positive, and neutrality and is on a [-1,1] scale where -1 is most negative and +1 is most positive.

Data Analysis

First, we examined contributions over time. Figure 1 shows a visualization of the total number of commits as a function of time for the top 90 repositories. Although a couple of the repositories date back to pre 2004, the majority of commits for many of the top repositories are from 2010 onwards (MSR dataset only goes up to mid-2013). To visualize the main contribution time more, Figure 2 shows a snapshot starting from 2010 to the mid-2013 of total commits over time.

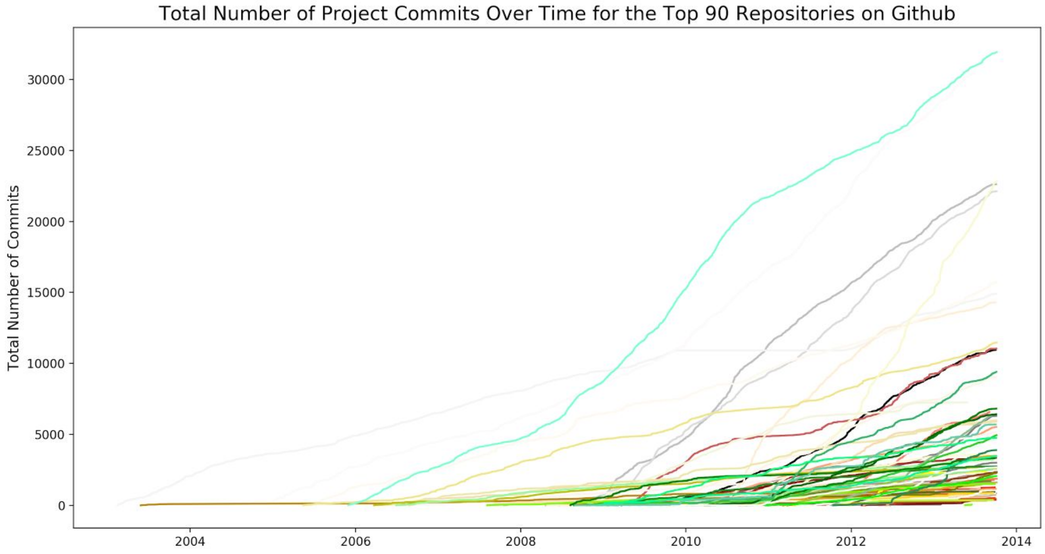

Figure 1: Total Github Commits Over Time

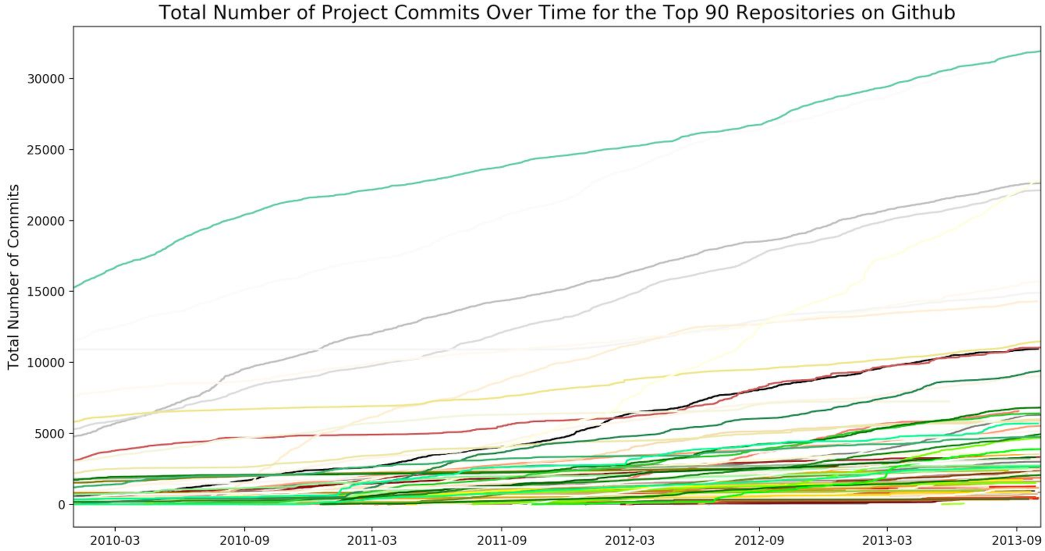

Figure 2: Total Github Commits Over Time from 2010-2013

Next, we looked at pull request comment sentiment. The pull request comment history dataset has fewer data points than the commit table and thus our dataset only goes from 2011 to mid-2013. In Figure 3 we depict total sentiment of pull request comments as a function of time for the top 90 repositories. Each repository is indicated by a different colored line. The total sentiment is calculated by adding the compound sentiment scores for each pull request comment.

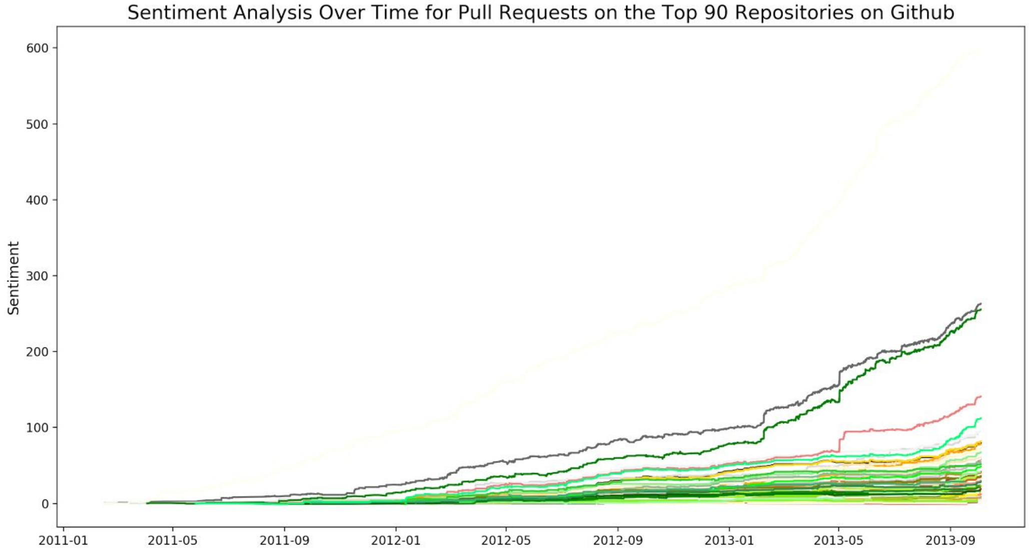

Figure 3: Total Github Pull Request Sentiment Over Time

Since almost every project's sentiment curve increases, overall sentiment is neutral to positive on the whole. We can see that the slope of total sentiment analysis is similar to the slope of commits over time. We also thought it was important to view average sentiment over time since the number of pull request comments can also affect the rate at which total sentiment increases over time. Figure 4 shows average sentiment over time.

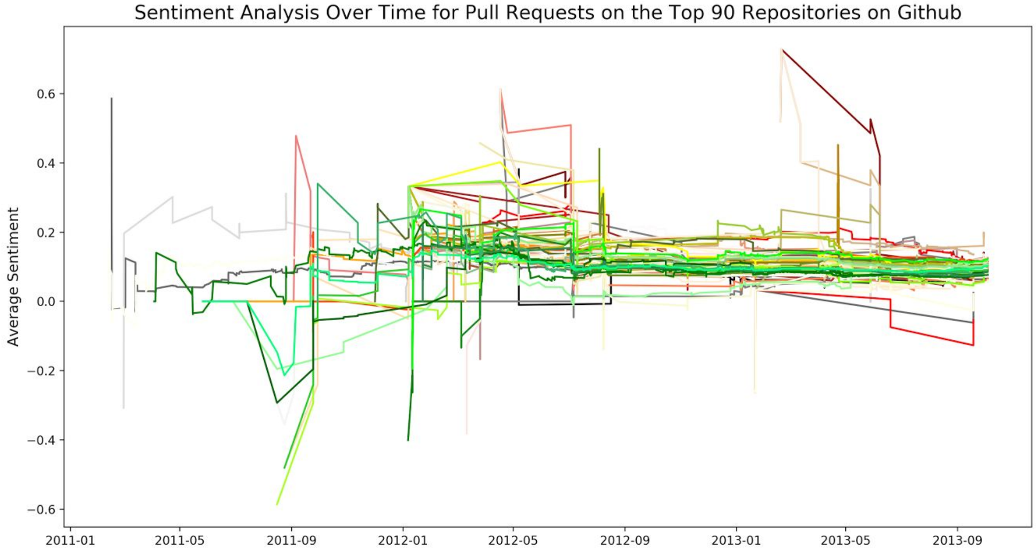

Figure 4: Average Github Pull Request Sentiment Over Time

Average sentiment is a lot less obvious to discern patterns from since the curve can jump rapidly especially when a repository has few pull requests.

Finally, to answer the main question of this paper, *if project sentiment is correlated with project contribution*, we plotted total project sentiment (normalized) and total project commits (normalized) over time for each of the 90 repositories. Below we show in Figure 5 the plotted data from some of the top 90 repositories. Neither of the authors have a strong statistics background so we were unable to analyze the actual correlation of the two curves with confidence intervals and p-values, however, our visualizations did provide valuable overall analysis.

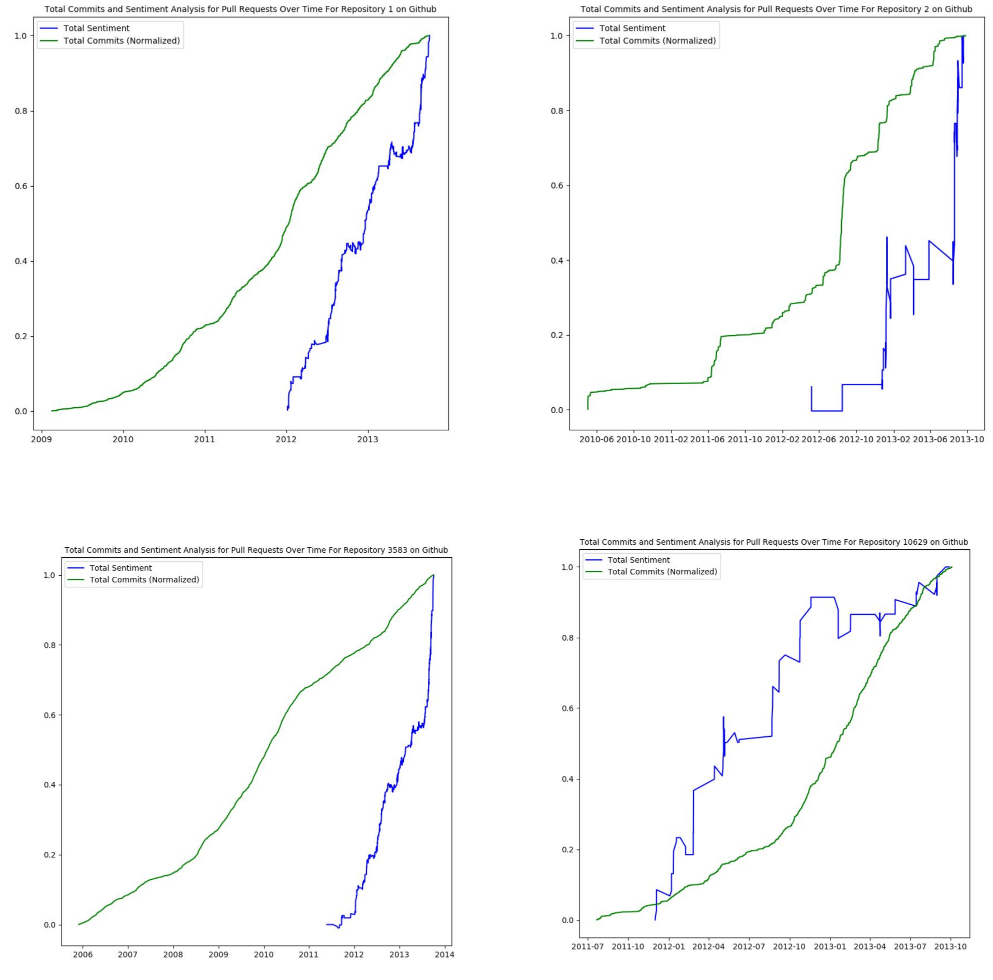

Figure 5: Total Github Comment Sentiment and Commit History (Normalized) Over Time

It is hard to discern any real relation from the data because of how noisy the sentiment analysis curves are. However, we see that both sentiment analysis and commits increase over time and if we plotted our curves with the same starting date they seem to be highly correlated (minus noise) in terms of their curve rates.

Reflection and Error Analysis

Using the dataset and the above methods allowed us to find the answer for the question, which is that both the sentiment of the comments and the number of commits increases over time. Hence, it indicates that the sentiment may have an influence on the total number of contributions. However, the lack of strong statistical background as we mentioned earlier limited our ability to determine the significance of the results for the sentiment analysis and contributions over time.

We believe that the answer of the question could have been expressed clearer using interval estimates on the observed data. In addition, the MRS dataset was very noisy. The pull request commenting data lacked a significant portion of dates and the amount of pull request data was variable from project to project. Nevertheless, even if the sentiment analysis scoring percentages were dense and in a closer compact, the correlation would have been extremely difficult to determine using observation only.

A huge source of error in our analysis could possibly be from the methodology. Assigning sentiment is a complicated task in of itself and can be prone to error. On a dataset of 4,200 tweets, Vader receives a F1-score of 0.96, however on 5,190 NY Times Editorial article snippets, Vader receives a F1-score of 0.55 [5]. Thus, it is apparent that depending on the dataset, the sentiment analysis error rate can vary widely.

To further explore Github pull request sentiment analysis accuracy, we analyzed some of the sentiments assigned to various comments. We found our system performed well at slightly to extremely positive comment analysis. Of the positive comments predicted (with compound score > 0.75) most seemed to have a positive tone (this analysis was done by human perception). However, our system poorly predicted negative comments. One reason could be the disconnect between discerning coding terminology and actual sentiment. For example, a comment that addresses *killing a connection* might have a positive tone overall, but the mention of *kill* multiple times confuses the sentiment analyzer into thinking it's a negative comment. Below are some examples of pull request comments and the sentiment percentages assigned to them. The examples show Vadar sentiment assignments performing well and poorly.

Comment	**Positive**	**Negative**	**Neutral**	**Compound**	**Human Classification**
great observation! yes, ideally progress would be reported after the bytes have been transferred. However, this would involve plumbing the callback down into the subtransport, as it is the	0.266	0.0	0.734	0.9134	Positive

component that knows when the bytes have actually been delivered.					
No, since this script is going into the git repository, it should be able to assume it has the PGP keys in that directory already. I just mean touching the user's personal PGP key library is probably a bad idea.	0.133	0.0	0.867	-0.6908	Negative
@davidfowl If we fail to parse the json response do we want to try reconnecting or just kill the connection? I'm thinking it might be wise to attempt to reconnect and let it fail out that way instead of pre-emptively killing it. Thoughts?	0.076	0.284	0.639	-0.9332	Neutral
Good point. Removing it a valid optimization and will not cause any trouble.	0.463	0.0	0.537	0.7755	Positive

Overall, we aimed to explore the factors that influence project contributions in an online environment. More specifically, in our paper we analyze project sentiment and its influence on project contribution. Although we cannot make any statistical conclusions of correlation, we did discern that both the commits and project sentiment increase over time at a similar rate suggesting a possible correlation between the two variables. More statistical analysis needs to be done before further conclusions can be made.

Division of Labor

Fatima worked on pull-request sentiment analysis and paper documentation. Natasha worked on contributions over time, data visualization and paper documentation.

References

1. Georgios Gousios: The GHTorrent dataset and tool suite. MSR 2014: 233-236
2. Stéfan van der Walt, S. Chris Colbert and Gaël Varoquaux. The NumPy Array: A Structure for Efficient Numerical Computation, Computing in Science & Engineering, 13, 22-30 (2011), DOI:10.1109/MCSE.2011.37

3. John D. Hunter. Matplotlib: A 2D Graphics Environment, Computing in Science & Engineering, 9, 90-95 (2007),DOI:10.1109/MCSE.2007.55
4. Steven Bird, Ewan Klein, and Edward Loper (2009). Natural Language Processing with Python. O'Reilly Media Inc.
5. Hutto, C.J. & Gilbert, E.E. (2014). VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. Eighth International Conference on Weblogs and Social Media (ICWSM-14). Ann Arbor, MI, June 2014.
6. Bing Liu, Minqing Hu and Junsheng Cheng. "Opinion Observer: Analyzing and Comparing Opinions on the Web." Proceedings of the 14th International World Wide Web conference (WWW-2005), May 10-14, 2005, Chiba, Japan.
7. Jurafsky, Dan, and James H. Martin. *Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition*. Dorling Kindersley Pvt, Ltd., 2014.