GitHub Commit Sentiment Analysis

Sourabh Shetty

CS6704

Introduction

The open nature of GitHub allows us to view projects across multiple languages, countries, and organizations. All of this data is available online in datasets that have already been generated by previous researchers. It would be very fascinating to analyze the data and derive any possible correlations that we may find here.

The Problem

In particular, I am interested in looking at GitHub commits. Code is written in programming languages and you cannot derive much insight about the thoughts and sentiments of the developer by simply looking at the code changes. Commit messages, however, are more personal to the developer and we might be able to find a lot more in there to analyze about the developer.

We can assume that a majority of the commit messages will have a neutral sentiment, since most commit messages tend to follow a fixed pattern simply describing the changes. However, it’s that small percentage of non-neutral commit messages that are the mystery.

Why Is It Important?

This problem is important because it would lead to insights about the state of mind of a developer. Organizations do a lot of research into understanding how developers think so as to improve efficiency. I believe understanding how they commit code is a key metric for which further research is needed.

My Solution

The neutral messages are not the problem and are not as interesting. The non-neutral commit messages, however, can lead to interesting questions being posted, and so it is these that I intend to study. For positive sentiment messages, questions could arise like whether they occur closer to the end of a project or to the beginning. For negative sentiment messages, questions could arise whether these tend to occur close to bug fix commits.

There are various other factors too that could potentially affect the sentiment associated with a commit message. For example, perhaps a developer working on a fix overtime and overnight could be more frustrated while committing the code. In such a case the time of the commit could be an interesting metric to study.

My work here would build on previous work already done, and thus would be comprehensive in terms of breadth. I believe my work could be used to develop some key insights into commit behavior, and also be used in future research in these topics.

Related Work

This is a popular topic and thus a lot of research has already been done on this.

“Analyzing Developer Sentiment in Commit Logs” by Vinayak Sinha, Alina Lazar, and Bonita Sharif published in 2016 IEEE/ACM 13th Working Conference on Mining Software Repositories deals with a similar topic.

They analyze a lot of the similar problems, and have a list of future work that I intent to look into while doing my analysis. One key point they haven’t looked into is whether there is some correlation between file change (addition, deletion, or modification of a file) and the sentiment of the resulting commit message.

An older paper, “Sentiment Analysis of Commit Comments in GitHub: An Empirical Study” by Emitza Guzman, David Azócar, and Yang Li, and published in MSR 2014 Proceedings of the 11th Working Conference on Mining Software Repositories, sees the correlation between programming language and the sentiment of the commit messages. In it they observed that Java commit messages tend to be the most negative. The idea of using programming language as a metric is also a good idea.

The paper also tries to see if there is some correlation between the number of stars and the commit messages, since they see the stars as an indication of “liking” as on social media platforms. While they didn’t find any correlation, it is still an interesting idea to including stars on a project as a metric.

This leads to a paper titled “Mining Communication Patterns in Software Development: A GitHub Analysis” by M. Ortu, T. Hall, M. Marchesi, R. Tonelli, D. Bowes, and G. Destefanis. While the paper deals with both commits as well as issues, focusing on just the commits will be very helpful. They look at the correlation between messages and the attractiveness of a project to newcomers, and whether more positive and polite messages help. This is a better metric than just looking at the stars as the previous paper suggested as well.