Humor Detection in GitHub Commit Messages using Sentiment Analysis Techniques

|  |  |  |
| --- | --- | --- |
| Katy Herrick  New York University  NY, NY 10005  kmh530@nyu.edu | Andrew Klingelhofer  New York University  NY, NY 10005  ack377@nyu.edu | Deniz Parlar  New York University  NY, NY 10005  dep295@nyu.edu |

|  |  |
| --- | --- |
|  |  |

Abstract

We introduce a targeted approach for automatically detecting humor among GitHub commit messages. The commits of a single GitHub project repository are classified as either “funny” or “serious” with respect to which term

1. Introduction

GitHub is an online service used mostly by computer programmers where users create online repositories to store project files (mostly text documents and computer programs). The action of uploading a new file or making changes to an existing file is called making a *commit,* and each commit is required to be accompanied by a short *commit message*. The commit message is usually between 50 and 120 characters in length and typically explains the changes made. However, because there are no enforced restrictions on what can be entered as a commit message, the content can act as an indicator of the personality and mentality of the programmer behind it.

We propose a method of classifying commit messages as “serious” (i.e. explaining changes made) or “funny” (i.e. anything else) whereby we repurpose traditional sentiment analysis techniques to train a classifier to detect instances of “funny” words and phrases, and then augment the classifier with the ability to identify typos and language specific to the GitHub project.

While this application of humor detection is unlikely to be of great impact practically, it does provide a way for programmers to filter down their (and their collaborators’) commits to find particularly stressful or productive periods in the development of their projects. Furthermore, this application could be used to help determine if one has the correct “personality fit” to join a new team. However, one would be advised not to take the results of this application too seriously.

There has been an amount of research in the area of humor detection.

* 1. Defining Humor

[subsection 1.1 stub]

* 1. Characteristics of Commits

[subsection 1.2 stub]

1. Approach

Our approach is to…

* 1. Capitalization

[subsection 2.1 stub]

* 1. Out-Of-Vocabulary Terms

[subsection 2.2 stub]

* 1. Typos

[subsection 2.3 stub]

1. Machine Learning Methods

[section 3 stub]

[section 3 stub]

* 1. Baseline

[subsection 3.1 stub]

* 1. Naïve Bayes

[subsection 3.2 stub]

1. Evaluation

[section 2 stub]

[section 2 stub]

* 1. Subsection

[subsection 2.1 stub]

1. Conclusion

[conclusion stub]

Reference

[reference stub]

[reference stub]

Reference

[reference stub]

[reference stub] Reference

[reference stub]

[reference stub]

f