Analyzing Trends in Open Source Software Contributions

Andrew Casner
University of Colorado, Boulder
andrew.casner@colorado.edu

Oliver Collins
University of Colorado, Boulder
oliver.collins@colorado.edu

Carl Cortright
University of Colorado, Boulder
carl.cortright@colorado.edu

Shubha Swamy
University of Colorado, Boulder
shubha.swamy@colorado.edu

1. Problem Statement

Our primary goal of this project is to gain insights about open source software. We will accomplish this goal by mining data from a dataset about open source projects. We will focus our work on four main areas: tracking trends in programming languages, analyzing how popular repositories have changed over time, how contributions to those repositories have changed over time, and also monitoring repository life cycles.

2. Literature Survey

There has been a significant body of research done on patterns in open source contributions. Every year Stack Overflow, a popular site for asking and answering coding questions, does a poll of the developers on their platform [1]. This research is used to track popular languages and libraries developers are using on the platform. Other research focuses on the potential impacts of research on open source software on the broader computer science community [2]. This research can help act as a guide as we decide what areas of open source development are important to focus on in our data mining project. The majority of the research in this area has focused specifically on Github repositories, attempting to determine the influence of any given project [3]. In our work, with the dataset we have access to, we will attempt to extend this research to all open source projects independent of where they are hosted or built. Other research by the Apache foundation has shown that the majority of open source contributions are made in a way that is not collaborative but instead driven by individual developers [4].

3. Proposed Work

Since this dataset is already in excellent condition, not much will have to be performed on the dataset. However, some necessary data cleansing will need to be completed such as scrubbing the dataset to remove null values and synchronizing time zones. Once these simple tasks are complete, our team will perform more advanced data preprocessing techniques such as matching a unique user across multiple package managers.

Once our team has appropriately cleaned our data, we will then decided which patterns answer the questions asked of the dataset. Those will be used to create visualizations, such as bar plots and graphs to show the trends found in the given data. Using the data available, we will construct a dependency graph of all of the dependencies in the entire open source ecosystem. We will then conduct an eigenvector analysis on this graph to find the most influential projects. This same analysis can then be done on language-specific sub-graphs to find the most influential projects in each language.

Finally, we will create a write up of what we have learned and possibly publish it online in a way that is consistent with the open source mentality that we are studying. It is essential to our team that what we learn about the open source community can be freely and openly shared. Our write up is also aligned with the license that the libraries.io dataset is under.

4. Dataset

The dataset we will be using for our project comes from a company called libraries.io. The purpose of libraries.io is to monitor open source projects to help developers better understand dependencies for their projects.

The dataset itself contains 311 million data points from 34 package managers and three source code repositories, including npm, GitHub, PyPi, RubyGems, Maven, Bower, and other large, language-specific package managers. With this breadth, libraries.io can track over 2.7 million unique open source packages, spanning 31 million repositories, tracking 161 million dependencies between them [5]. With this large of a dataset, we expect to have many different variables of interest that we can mine.

The dataset comes packaged in several large CSV formatted files. The main file that we will be mining is the projects database. This file contains all of the individual projects libraries.io is tracking. This file also has vital attributes for each project including language, status (active, depreciated, etc.), dependent projects count and more. Other important datasets include the dependencies CSV, which has detailed information on interdependencies between projects. Using this module we can build a dependencies graph, and mine information about which projects are most important in the open source ecosystem.

5. Evaluation Methods

We will be gathering various conclusions from our dataset. Our project focuses on four areas: tracking trends in programming languages, analyzing how popular repositories have changed over time, how contributions to those repositories have changed over time, and also tracking repository life cycles. We will need to utilize various evaluation methods to draw insights about those topics. To track trends in programming languages over time, we will analyze the frequency of popular programming languages used in various open source projects and how that has shifted over time. To analyze how popular repositories have changed over time, we will investigate how project dependencies of various open source projects have changed over time. To analyze how contributions to repositories have changed over time, we will first narrow down our data range to top contributed repositories, and then we will analyze the rate at which the contributions to those repositories have changed over time. Through this, we will also be able to conclude information about repository life cycles. By looking at trends and dependencies of various repositories over time, we will be able to conclude the repository's life cycle.

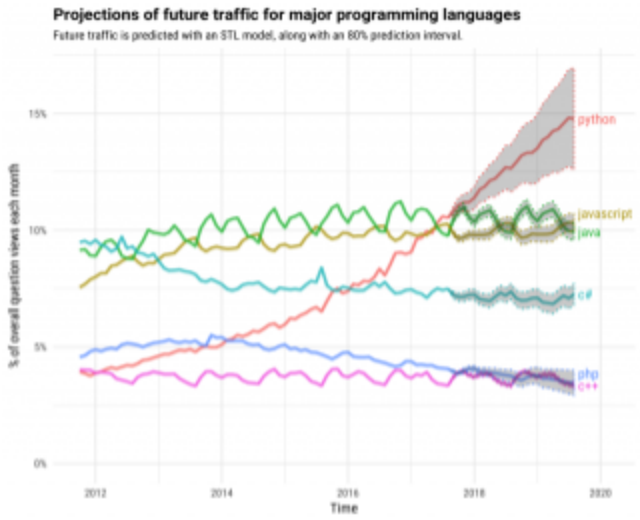

Fig 1. Predictions of future traffic for major programming languages. Graph from StackOverflow, Web. 8 Sept. 2017. n. pag.

In addition to analyzing our data in regards to the four main areas we will be focusing on, we will also compare our results to conclusions found in prior work (2).

6. Tools

We will be using several tools to mine and analyze the dataset to reach our conclusions about open source software. The primary programming language in this project will be Python. We will be using various Python libraries to analyze, parse, and present our data. One of the tools we aim to use, Pandas, will be

used for most of our data analysis. As for our computational and statistical analysis, we will be using SciPy and NumPy. Since our team will be presenting our findings through data visualizations, we will be using a plethora of different libraries including Matplotlib, Bokeh, graph-tool, Seaborn, and several others. Moreover, lastly, for a lot of our numerical simulations, statistical modeling, data visualization, as well as a way to keep our code clean and organized we will be encapsulating our code in Jupyter Notebooks.

7. Milestones

We will be following the same timeline as outlined in the project description in regards to our milestones.

Milestone	Due Date
Proposal Presentation	February 27th
Proposal Paper	March 6th
Progress Report	*April 10th*
Final Presentation	*April 23rd*
Final Paper	*May 1*

8. Summary of Peer Review Session

While we did not receive any questions/comments from our peers after our presentation of our proposed project, we were advised to change the way we evaluate our project. We have since revised our evaluation, which we have included in the Evaluation Methods section above.

However, while listening and hearing from other project proposals, we gained insight on ways to tackle our analysis and different approaches.

References

[1] "Stack Overflow Developer Survey 2017." Stack Overflow, insights.stackoverflow.com/survey/2017.

[2] Scacchi, Walt. "The Future of Research in Free/Open Source Software Development ." 2010.

[3] Hu, Yan, et al. "Influence Analysis of Github Repositories." SpringerPlus, Springer International Publishing, 5 Aug. 2016, www.ncbi.nlm.nih.gov/pmc/articles/PMC4975729/.

[4] Chełkowski, Tadeusz, et al. "Inequalities in Open Source Software Development: Analysis of Contributor's Commits in Apache Software Foundation Projects." PLOS ONE, Public Library of Science, 20 Apr. 2016, journals.plos.org/plosone/article?id=10.1371%2Fjournal.pone.0152976.

[5] libraries.io/data.