Language Communities on GitHub

Puja Jena, Hayeon Kim, Frances Tso, Sharon Yu, Jake Zimmerman

February 28, 2017

Software development is social

- Software is built collaboratively
- GitHub is a platform for “social coding”
- GitHub users collaborate on many tasks:
  - reporting bugs
  - discussing future changes
  - reviewing code quality

Programming languages create communities

- Each project has a primary language
- Same language? Same habits & customs
- Language communities accumulate stereotypes
  - C, C++? → "old-school"
  - JavaScript, Ruby? → "hip"
  - SML, Haskell? → "academic"

GitHub has been studied extensively

- Which projects and languages are popular?
- Which languages are used frequently?
- Using “interest graphs” to gauge which topics are interesting
- How does transparency affect software development?

We wish to answer:

How does social activity on GitHub vary by programming language community?

Along the way, we’ll look at questions like these…

- Which emojis are most common in this community?
- Are emoji reactions on threads common?
- Are there vernacular trends (acronyms, abbreviation, phrasings) in this community?

- Which other language communities does this community talk about?

- Is this community characterized by
    - a low number of frequently active contributors?
    - a high number of infrequently active contributors?
    - something in between?

- Do projects in this community have
    - more pull requests than issues?
    - more things closed than open?
- Are discussions resolved quickly, or do they drag on?

- Are contributions primarily
    - during the day?
    - in the evening?
    - late at night?

Beyond “just measuring popularity”

- Look at projects in each community which have **comparable numbers of stars** (stratify)
- Sample “canonical representatives” of a language community

Presenting our findings

- Visualize a handful of the answers to these questions
- Critically analyze our findings

Who cares?

- Maintainers of large open source projects
    - Trying to ensure positive community for large number of people
- Hobbyist programmers
    - Trying to deliver small- to medium-sized product for specific group
- People looking to join a new community
    - Want to understand how that community is different from ones they're familiar with

What could go wrong?

- Hard to find good data
- Data gives no “interesting” insights
- Visualizations don’t communicate findings well

Deliverables

- Midterm
  - Interview for people's preconceptions of GitHub
  - Sample dataset representative of final dataset
  - Initial prototypes of visualization graphics
- Final
  - Analysis based on larger dataset
  - Have graphics visualizing data and our analysis