

Measuring Innovation Using Novel Sources of Data

Project Goal

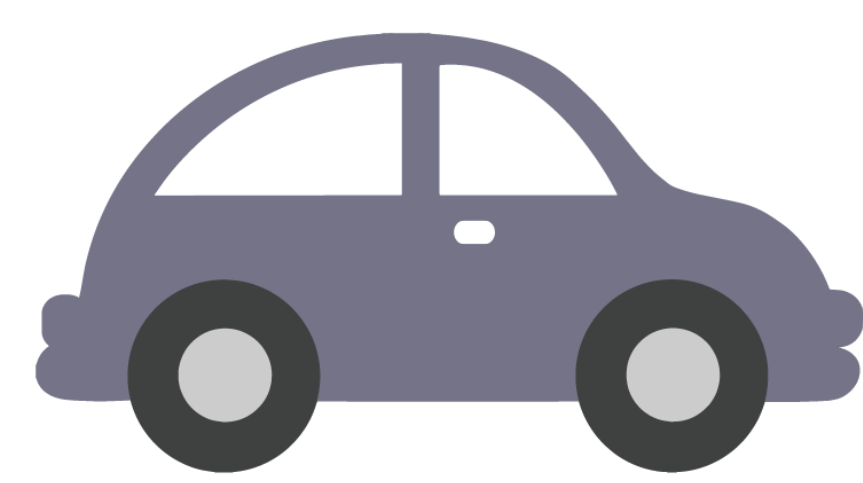
To identify potential sources of data

- To measure the production, diffusion, and impact of open-source software that is freely available on the Internet.
- To develop useful statistics to measure the value of open source software, as well as the linkages to institutions and sectors.

INVENTION VS INNOVATION [1]



INVENTION When a new original product is created.



INNOVATION When a new product is created or a previous product is improved upon to meet the current market needs.

Current Practice

- Measurements of innovation tend to rely on survey data, patent issues, trademarks approvals, intangible asset data, or estimates of total factor productivity growth. [2]
- Survey data are incomplete, and patent data fail to capture innovation that is freely available to the public.

Challenges in Measuring Open-Source Software Innovation

Open-source software is difficult to capture using conventional methods of measuring innovation.

- Software companies can be short-lived and volatile, hence, surveys will have coverage issues of the software industry.
- Open-source software is licensed, but not patented.

There is a need to search for novel sources of data, such as software licenses, users and contributors, or updated versions.

Open Source Software

- Definition:** software with a license that the copyright holder makes the source-code available to users to view, alter, & share.
- Some open source software may require software that draws from the source code to be distributed under the same conditions as the original software license, a practice known as “copyleft.”
- We explored 12 subject areas including Business, Education, Web, Data, etc.; carefully researching an example of each and evaluating documented funding, social media usage, number of contributors, etc.



- The next step was to go to the avid users and contributors themselves.

Interview Responses

What is the incentive to make or use open source software?

Daniel Chen: “Knowing your work is completely public for anyone to view and audit is a good incentive to not write bad code. Open source, by its very nature, makes cost a non-issue, making it easier to adopt than proprietary software. The more users who use a tool, the more people are potentially testing your code, thus over time making the codebase more robust.”

Brian Goode: “Now that computing has become more widespread, it is no longer a niche market. This makes it impossible to corner a market, because someone with a need will develop software locally for their own personal use, and afterwards make the code public. Coders may not expect anything in return, but the internet has made sharing code so easy that it involves little additional work for authors to share their software freely with the online community.”

What is the process for contributing to an open source of software?

Daniel Chen: “Find where the code is hosted (a popular service is github), and begin reading the documentation, installation guide, and quick start instructions. Many online hosting services have a bug report/issues tab to start looking at known problems. They may also have an IRC/Gittr chatroom or even a mailing list where people can ask the community where to start.”



DANIEL CHEN
PH.D CANDIDATE AT VIRGINIA BIOMPLEXITY INSTITUTE
M.S. EPIDEMIOLOGY COLUMBIA UNIVERSITY
ACTIVE PYTHON, R AND GIT USER



BRIAN J GOODE PH.D
RESEARCH SCIENTIST
VIRGINIA TECH DISCOVERY ANALYTICS CENTER
ACTIVE PYTHON, R AND GIT USER

Methodology

Research Questions

- What are the incentives to make or use open-source software?
- What areas and fields does open-source software dominate vs. the areas and fields that proprietary software dominate?
- Is open source software driven more by business needs or individual needs?
- Where does most funding come from for open source software?
- How commercially viable are open-source software versus proprietary software?

Define Key Words

“OPEN SOURCE SOFTWARE”
“PLATFORM” “BUGS”
“FUNDING SOURCES” “CONTRIBUTE”
“SUCCESS” “OPEN VS PROPRIETARY”
“OPEN VS FREE” “MEASUREMENT”
“VERSIONS” “COLLABORATION”
“OPEN SOURCE” “INNOVATION”

Conduct Interviews

Literature Review Process

Conduct Title Search

Read through article abstracts of collected articles and sort by most useful

GREEN: A ‘superior selection’ which relates to research question(s) and matches interests

YELLOW: A moderate selection’ which relates to research question(s) but does not match interests

RED: A ‘poor selection’ which does not relate to research question(s) and does not match interests

Preliminary Findings

Main Topics

- Incentives** - user need and user reputation motivate programmers to make their code open-source, or contribute to open-source projects.
- Collaboration** - knowledge commons, crowd-sourcing, and online platforms should be used to overcome challenges of collaboration and to encourage collaborative software design. [3]
- Adoption** - the main industries or fields of study that have adopted open-source software are web development, data science, and academic research.

Potential Measures

- Version Counts**
 - Shows how software is developing
- Contribution Counts**
 - Shows how interested developers are in growing the product
- Social Media Usage**
 - Shows number of users and the response of the community to the product