Can we scale open-source development by removing the need for iconic and self-sacrificing leaders?

Simon Carbonnelle
ISP seminar
2nd of October 2019
Users have the rights to study, change, and distribute the software to anyone and for any purpose.

The software's publisher or another person retains intellectual property rights.
1. Understand the problems of open-source

2. Explore how we can rebalance forces
Why do individuals participate?

**Organizational culture**: the vision/values and social aspects of an organization

**Open-source** attracts members mainly through the cultural aspect of work.

**Private sector** attracts members mainly through salaries/compensations.

Which poses huge **constraints** for open-source:

- **Vision alignment or politics (Problem 1)**
  The organization needs a transparent and shared vision
- **Quality of social interactions (Problem 2)**
  Interactions between members need to be positive and stimulating
Vision alignment or politics

Examples from **open-source**

“The FFmpeg development model was fairly simple: the biggest FFmpeg contributor (Michael Niedermayer, see stats) was the "self-proclaimed" maintainer.”

“The main complaints I saw mentioned and remember from that time were about the project leader. It was all about "I do and commit whatever I want to", "I don't care about cosmetics", "I don't want to discuss with you in real life, mails are good enough".”

**Result:** Community splitted into two forks... frustrating contributors and users
Vision alignment or politics
Examples from open-source

Why Neovim is Better than Vim

“The codebase is atrocious. The plugin API is cumbersome and restrictive. The dev community is apathetic. The benevolent dictator is averse to change.”

“Speaking of Bram Moolenaar: His merge criteria are inscrutable. Some patches he ignores. Some, he attacks. Others, he merges.”

Result: Community splitted into two forks...
“The problem is that the way you talk to people drive contributors away. This is my case. And I assume that this was also the case of Jason.

You may be doing fantastic technical work, I won’t judge that. But in the end, it remains that non-ore devs have to work with you to contribute. And this is my opinion on the matter: working with you is not fun. And this is a problem because working for a free projet has to be fun to attract new contributors.”
Vision alignment + social interactions
Examples from open-source

After governance breakdown, Node.js leaders fight for its survival

The tumultuous events this week in the Node.js community were a long time coming. Insiders tell ZDNet how the community is trying to repair — and prepare for what's to come.

"Rod's behavior was non-collaborative, consistent, and unapologetic," said Williams

The resignations followed a single event -- a vote that failed to remove a former director, a longstanding member of the community, from the leadership group.

Result: a third of the technical steering committee quits, a new fork is launched with core members

"I'm confident that the individuals who voted on this didn't quite realize how big of a thing this was going to be"
Vision alignment or politics

Examples from private sector

Inside Google’s Civil War

Some employees say Google is losing touch with its “Don’t be evil” motto. What happens when an empowered tech workforce rebels?

By Beth Kowitt
May 17, 2019

February 2018 PROJECT MAVEN LEAKS: Most employees learn for the first time that the Pentagon was using the company’s A.I. to analyze drone footage.

August 2018 DRAGONFLY LEAKS: The Intercept reports that Google is working on a censored search engine in China; it’s the first time most employees are hearing of the project.

Results: “Jack Poulson says he was the sixth or seventh employee to cite Dragonfly as a reason for quitting.”

Out of 100,000 employees!!!
Vision alignment or politics
Examples from private sector

The New York Times

Dozens at Facebook Unite to Challenge Its ‘Intolerant’ Liberal Culture

“The activity is a rare sign of organized dissent within Facebook”
1. Understand why open-source is losing

- politics

2. Explore how we can rebalance forces

- social interactions
How to make use of the available workforce?

Open-source contributors are mainly hobbyists, and demand flexible and lightweight workloads...

... but **full-timers are needed** in practice when an organization scales.

*(Problem 3)*
Not too long ago I started an open source project called fastlane. Just a month after publishing, it had 1000+ stars on GitHub and was beginning to get used by lots of serious tech companies around the world. Soon I was the sole maintainer of a project getting 10+ PRs/day, and spending 8+hrs/day reviewing PRs and replying to questions on GitHub.
Flexible and lightweight workloads
Examples from open-source

"People who maintain the project – the core team – these are who are responsible for planning, QA, code review, releases, they are the bottleneck."

"So there are enough people who write code. But it gets painfully long to get the code reviewed and merged."
Flexible and lightweight workloads
Examples from open-source

[FFmpeg-devel] FFmpeg's future and resigning as leader

Michael Niedermayer michael@niedermayer.cc
Fri Jul 31 13:53:04 CEST 2015

“Indeed I fully admit the work and pressure caused by the merges is a main reason for my resignation.”
Flexible and lightweight workloads
Examples from open-source

“Working day jobs, maintainers today can struggle to find the time to fix critical bugs, all the while facing incessant demands from users requesting free support on GitHub. Maintainer burnout is a monstrous challenge.”

A complete report: “Roads and Bridges: The Unseen Labor Behind Our Digital Infrastructure”
1. Understand why open-source is losing

2. Explore how we can rebalance forces
Current practical solutions

Elaborated political systems

How Do Open Source Communities Govern Themselves?
Submitted by rfay on Mon, 2012-03-05 20:41

- Technical steering committee
- Community committee/ Community Council
- Elections (representative democracy)
- Constitution
- Code of Conduct/ Social Contracts
Current practical solutions

Funding

“open your finances to your community”

“The best way for artists and creators to get sustainable income and connect with their fans”
Popular “theoretical” solutions
Towards radically different forms of organization

● Decentralization, DAO’s (blockchain-based)
  “An organization represented by rules encoded as a computer program that is transparent, controlled by shareholders and not influenced by a central government.”

● Horizontal organizations, holacracy, sociocracy, ...
  “An organizational structure with few or no levels of middle management between staff and executives”

Not just about open-source!
Group decision making
From theory to practice?
My best guess

Tools that increase our ability to cooperate on intellectual tasks at scale are a crucial ingredient.

Getting better at thinking together

How? We can get there by augmenting our communication protocols with:
1) **Structure**: the relation between ideas/pieces of information is explicit
2) **Value-based feeds**: participants can share the value they confer to content
### Structure

#### Examples

Arranging events through e-mail is a pain...

... Doodle solves the problem by offering a **structured** way to communicate around this task

(Framadate is a nice open-source alternative)
Value-based feeds

Examples

Reddit (Machine Learning community)
Value-based feeds

Examples

Reddit (Machine Learning community)

Aren't these comments beautiful? :)

[v] Tackling Climate Change with Machine Learning - video & blog post summary

This is fantastic. I'll definitely have to take a deep dive into this.

[jeff] 3 points · 1 day ago

Thank you from the bottom of my heart. I already work on ML stuff, but ask myself every day how I can make my work potentially useful for saving the planet or whether it's time to go do something that's more directly useful for that (and what that would be - if I knew for sure, I likely would have already been knocking on their door). I've seen this paper before, but lacked the patience to read through it. It's wonderful to see it being put in a more digestible form.

[paul_read_it] 2 point · 17 hours ago

Thank you for your kind words! Looking forward to find out what you'll be working on!

[bhoreffect] 2 points · 2 days ago

Thanks for posting this. Wrapping up my dissertation covering items 1-3 lumping it together as "civil infrastructure".

Will definitely share these links with my classically trained engineering friends, as ML concepts and the increased availability of municipal/civil data is only now just starting to percolate out to traditional/physical PE type work.

[LevishManatee] 1 point · 2 days ago

Thank you for posting this, I am fascinated.
Value-based feeds

Examples

Reddit (Machine Learning community)

Post by u/ledyaorb 7 months ago

[R] A Generalization Theory of Gradient Descent for Learning Over-parameterized Deep ReLU Networks

enri1c.org/abs/19...

Research

cianmodan 12 points · 7 months ago

The paper is 50 pages, so I might have missed something. But from the highlighting in the text, Theorem 1.1 resp. 4.4 is the central result. IMO, there are three aspects of the claim:

(i) By making the NN big enough, there exist parameter sets which achieve very low population loss. Qualitatively, this is ensured by the universal approximation property, and they impose something stricter about the ground truth (Assumption 4.2)

(ii) A parameter set with low population error can be found based on a finite sample n of size as specified in the Theorem.

(iii) Gradient descent works just fine in this regime, meaning that the non-convexity of the empirical loss surface is not a deal-breaker.

I think there has been a series of paper with results similar to (iii) recently, relying on the over-parameterization m >> n. Putting this in context with the generalization aspect (ii) is nice.

However, the sample complexity implied by their result is rather bad, corresponding to a rate of convergence n^4(1/6), which is much slower than the standard rate n^(1/2). Also, there is no quantitative evaluation of their generalization bound. Hence, it can be hardly called a “non-vacuous generalization bound”; in contrast to the cited work of Dzgust & Roi which yields an upper bound on MNIST of ~18% IIHC (Disclaimer: I'm not an author). Also, “explaining the good generalization performance” might be overselling the result regarding the sample complexity.

Another comment: in Remark 4.7, the authors state that the ‘over-parameterization condition is independent ... of n’, yet they require m > n^2.

Despite the slow rate of convergence, the result claims that the classification problem is “learnable” by gradient descent on NN models. I am not qualified to assess the novelty of this claim, but it seems to be a satisfactory qualitative result.
Structure + value-based feeds

Examples

Kialo

A total of 2700 arguments around the UBI question...

... All hierarchically structured and evaluated by peers.
How can we apply this to open-source?

1) Transform project management in a collective task
   ➔ all active contributors share power to merge code through voting

2) Improve cooperation around this (intellectual) task
How can collaborative thinking help open-source?

- **Politics**
  - Easier group decision making
  - Transparent deliberation

- **Social interactions**
  - Inappropriate language can be downvoted

- **Workloads**
  - Management task can be solved collaboratively
What’s missing?
What’s missing? Value-based feeds
What's missing? A structured way to communicate around code understanding

Understanding the code & getting started

The best strategy to better understand the code base is to pick something you want to change and start reading the code to figure out how it works. When in doubt, you can ask questions on the mailing list. It is perfectly okay if your pull requests aren't perfect; the community is always happy to help. As a volunteer project, things do sometimes get dropped, and it's totally fine to ping us if something has sat without a response for about two to four weeks.

So go ahead and pick something that annoys or confuses you about numpy; experiment with the code, hang around for discussions or go through the reference documents to try to fix it. Things will fall in place, and soon you'll have a pretty good understanding of the project as a whole. Good luck!
1. Understand why open-source is losing
2. Explore how we can rebalance forces
1. Understand why open-source is losing

   politics  
   social interactions  
   workloads  

2. Explore how we can rebalance forces
Thanks!
Thanks to...

Antoine, Arnould, Sam, Victor, Antoine, Maxime, Anne-Sophie, Ploum, my ex-flatmates (Mailtingpot), my flatmates (Cul-de-Sac), my family

...for the helpful discussions!

References for further readings....
definition open-source software: https://en.wikipedia.org/wiki/Open-source_software
definition proprietary software: https://en.wikipedia.org/wiki/Proprietary_software
some logo's: https://www.flaticon.com/packs/software-development-logos

Problems of open-source
libav-ffmpeg: https://www.theregister.co.uk/2015/08/05/ffmpeg_leader_steps_down/
http://blog.pkh.me/p/13-the-ffmpeg-libav-situation.html
neovim: https://geoff.greer.fm/2015/01/15/why-neovim-is-better-than-vim/
diaspora: https://discourse.diasporafoundation.org/t/project-management-at-diaspora-or-what-the-heck-am-i-doing-here/563/8
node.js: https://www.zdnet.com/article/after-governance-breakdown-node-js-leaders-fight-for-its-survival/
google: https://fortune.com/longform/inside-googles-civil-war/
diaspora: https://discourse.diasporafoundation.org/t/the-core-team-a-bottleneck-of-the-diaspora-development/588
felix kraus: https://krausefx.com/blog/scaling-open-source-communities
roads and bridges:
Solutions

open source governance  https://randyfay.com/content/how-do-open-source-communities-govern-themselves
patreon:  https://www.patreon.com/
opencollective:  https://opencollective.com/

Definition DAO:  https://en.wikipedia.org/wiki/Decentralized_autonomous_organization
aragon:  https://aragon.org/discover
colony:  https://colony.io/
daostack:  https://daostack.io/
democracy.earth:  https://democracy.earth/

Definition horizontal organization:  https://en.wikipedia.org/wiki/Flat_organization
Reinventing organizations:  https://www.reinventingorganizations.com/
Udn:  http://universite-du-nous.org/

framadate:  https://framadate.org/
trello:  https://trello.com/
stackoverflow:  https://stackoverflow.com/
reddit:  https://www.reddit.com/
kialo:  https://www.kialo.com/

Numpy’s github:  https://github.com/numpy/numpy
Value-based feeds

Examples

Stackoverflow

55 Answers!

Since the question refers to a single element, this code might be more suitable:

```javascript
// Checks css for display:[none|block], ignores visibility:[true|false]
$(element).is("visible");

// The same works with hidden
$(element).is("hidden");
```

Same as tweer's suggestion, but applied to a single element; and it matches the algorithm recommended in the jQuery FAQ