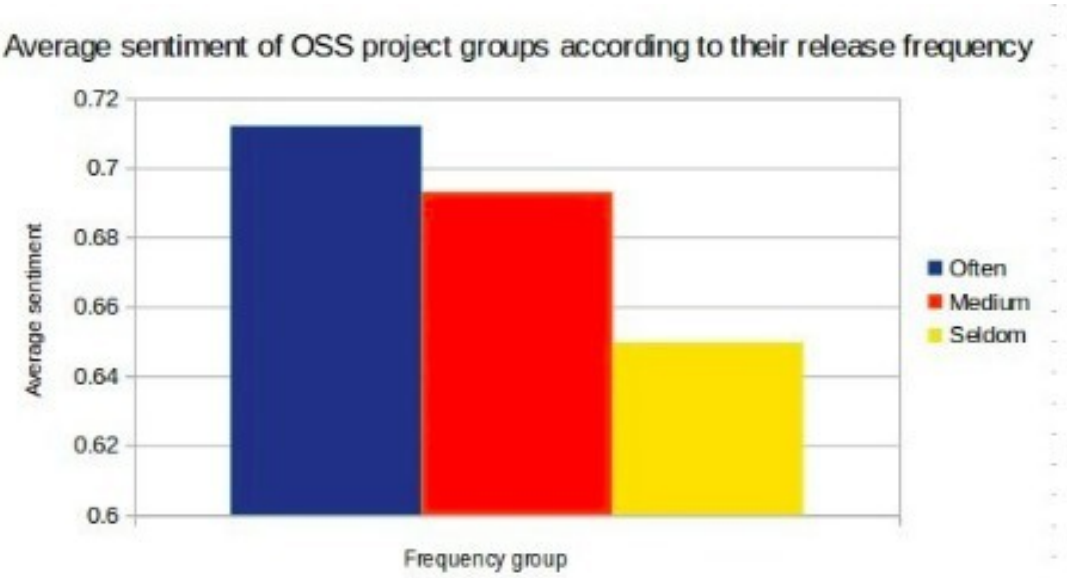
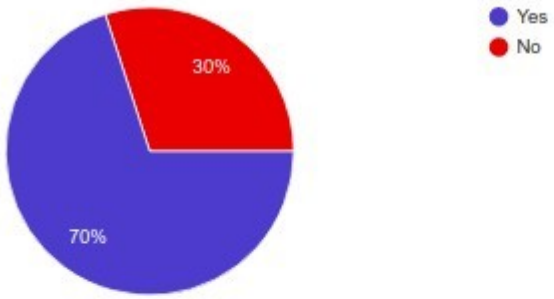
Appendix



Question 1: Would you expect following sentiment values?

10 odpovedí

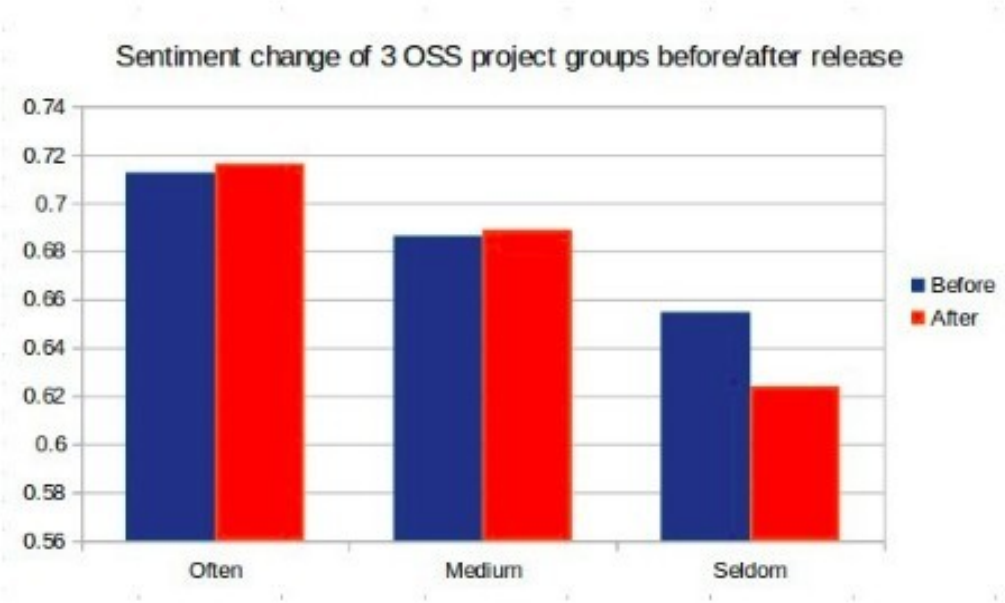


Additional remarks (if any):

4 odpovede

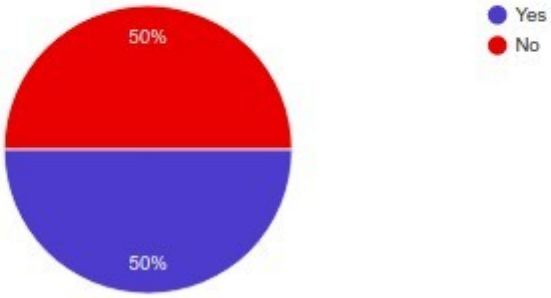
Releasing less then once per month doesn't seem negative to me. Think you'd need a bigger range like once every 1-3 months, once per 6-12 months etc.
I don't necessarily agree that this is a split yes or no answer. A project could release constantly, but if they're constantly shipping buggy code, it'll have a negative sentiment overall.
Idk how you're gonna normalize your data. React has many releases and I have no idea how people *feel* about it (other than obsessing over it for being easy, but having no clue how to use it in a big app.....). But Angular and Ember have fewer releases, but have great backwards compatibility and Ember has LTS releases which makes the sentiment very high / positive.
More often releases generally means that lots of patches for bugs are coming out, which leads to happier devs and a happier community

Question 2: I've measured sentiment 2 days before and 2 days after releases.



Question 2: I've measured sentiment 2 days before and 2 days after releases. Does is surprise you that it got significantly worse for projects with low(seldom) releasing frequency?

10 odpovedi



Additional remarks (if any):

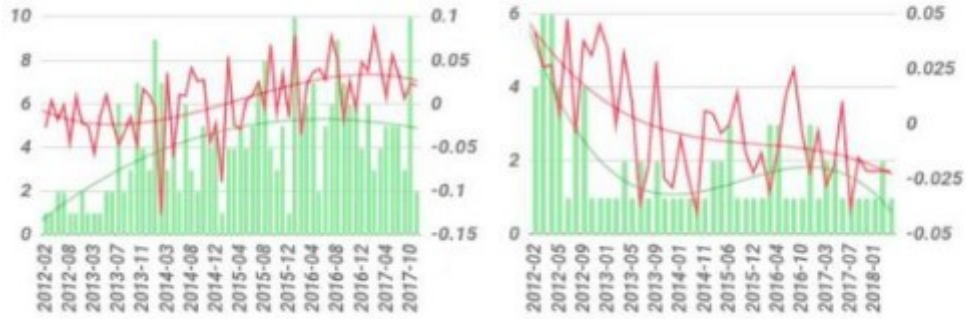
3 odpovede

Generally speaking I think it's difficult for most projects to keep every developer happy, and typically after a release people are reminded about the project and therefore will be more vocal. I would expect sentiment and volume of tweets to reduce as time passes from the release date.

Again, I don't know how this is normalized. But personally, for an upcoming ember release, there is some hype, and then once it comes out, people either get busy using it, or... idk. Also, with semver (something few projects do correctly), *major* releases remove deprecations, and don't add features.. it's intentionally boring.

it doesn't really, projects that don't update frequently are giving more time for buggy behavior to cement into the community and have code baked around it. This leads to breakage when the bugs are patched.

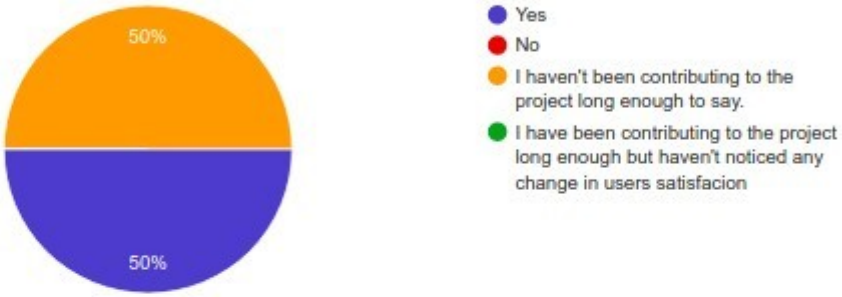
Question 3: Red line represents sentiment, green line represents release count



(a) EmberJS (b) Laravel

Question 3: Red line represents sentiment, green line represents release count per month. Find a project you've been contributing to lately. If you've been involved in the project over longer period of time, can you confirm the sentiment change over the years (red line)? Graphs start with the first release of the project and end around beginning of 2018.

10 odpovedí



Additional remarks (if any):

3 odpovede

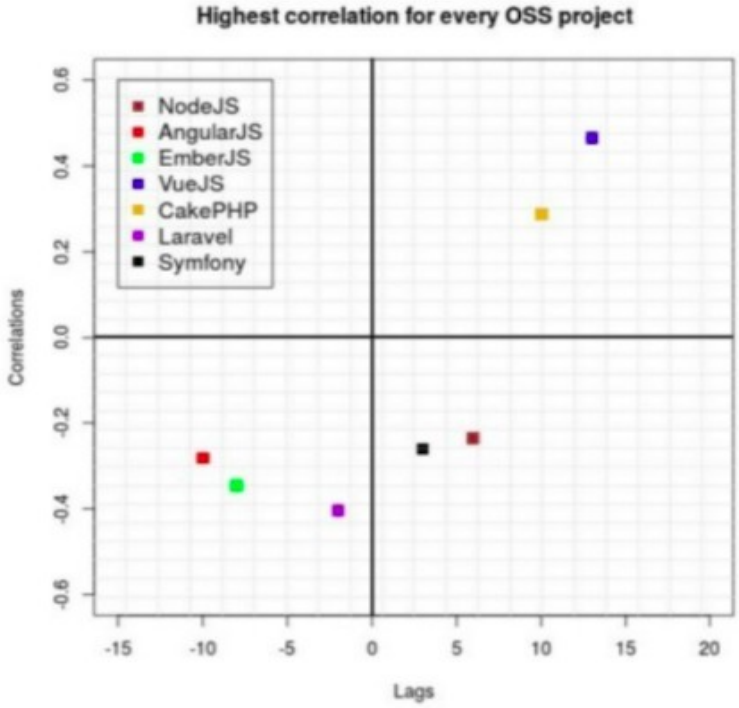
Is the higher value for sentiment good or bad? Can't really tell from the graph. That said, Laravel has been growing in popularity over the years and with that will have had an increased exposure to developers.

This seems kinda accurate for ember, I think. I'd be _very_ interested to see how this trend continues for 2018.

I haven't contributed to AngularJS (v1) / Angular (v4+), but the chart surprises me. A *lot* of people hated Angular 2.

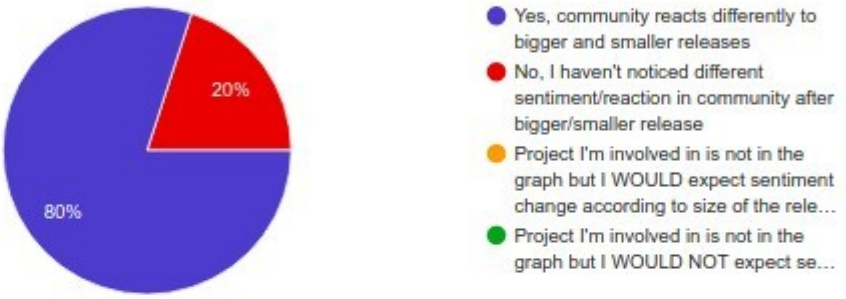
This is definitely roughly following the sentiment of the Ember community. The community dipped years back as Angular and others got more popular, then climbed as Angular and others went through many breaking changes. Recently sentiment has gotten worse again due to Ember "lagging" behind other frameworks in terms of features, though that seems to be changing again.

Commits:



Question 4: Calculating max cross-correlation points on the previous data series (sentiment + release count per month) gave me following results. Then I've changed the basic release count to commits count within releases (to see whether the size of releases plays a factor in sentiment) and the output changed significantly. Would you agree that your community reacts to bigger and smaller releases differently?

10 odpovedi



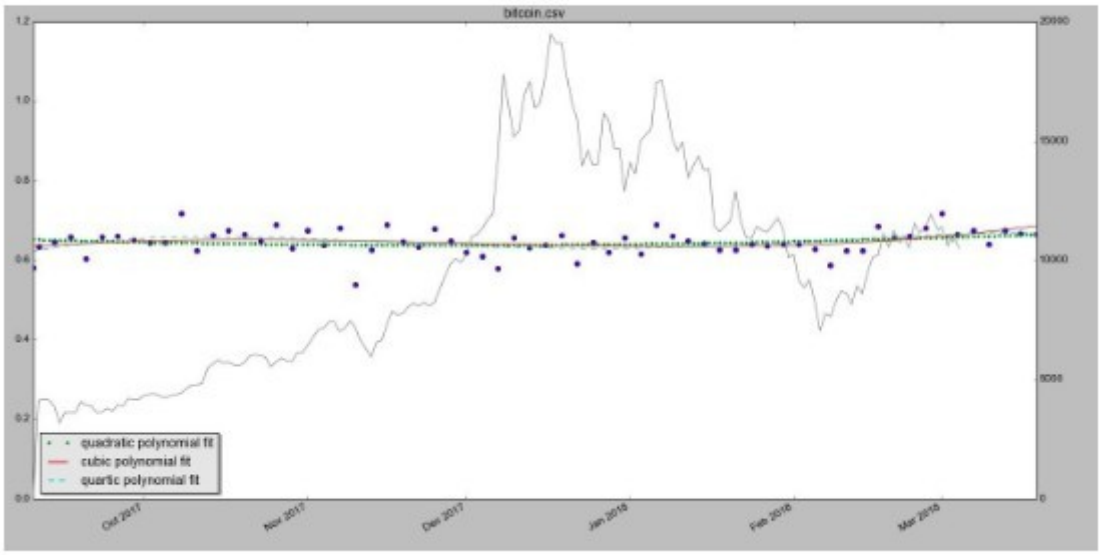
Additional remarks (if any):

2 odpovede

A high number of commits around a release could indicate that there are regressions (especially recently with ember-data :( ). But generally, I'm surprised that more commits for ember decreases sentiment. There has been amazing work done on ember recently, and a lot of positivity in general (at least that I've seen) – esp with #EmberJS2018 - https://www.google.com/search?q=emberjs2018+call+for+blog+posts&oq=emberjs2018+call+for+blog+posts&aqs=chrome..69i57.5334j0j7&sourceid=chrome&ie=UTF-8

I generally see higher sentiment on minor releases (every 6 weeks, include new features) than with patch releases (irregular, frequent, only fix bugs) because people are excited for new features.

Bonus question: Figure shows weekly sentiment measured on bitcoin tweets



Bonus question: Figure shows weekly sentiment measured on bitcoin tweets (blue data points). Describe in couple of sentences if and where you see the correlation between price development (grey line) and sentiment. You can point exact /approximate times.

4 odpovede

People seem to be more positive when bitcoin is stable and dislike drastic price changes in any direction

There was a huge amount of press speculation about the value of bitcoin which is where the spikes of sentiment for bitcoin appear. However when it started to drop in value the sentiment reduced.

it looks like people have no idea what they are talking about regardless of what happens with the price.

price goes up, sentiment goes down in one place and up in another – the inverse is also true.

Can't see any correlation

REDDIT: The table shows similarity score between git issue description and it's

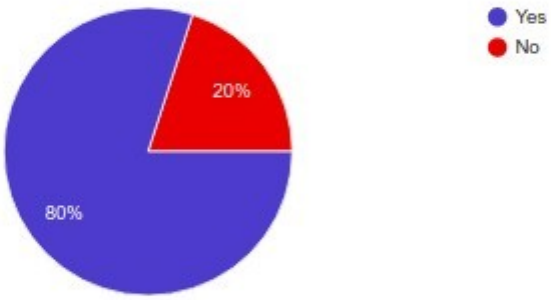
Framework	Bug comment	Whole discussion
NodeJS	0.447	0.507
AngularJS	0.306	0.57
VueJS	0.359	0.380

Table 0.19: Reddit NLTK similarity values

Part 2: Pairing reported Git issues with their related Stack/Reddit discussions

REDDIT: The table shows similarity score between git issue description and it's matching reddit discussion / particular comment which referenced the issue. Would you expect that the whole discussion shows often much higher similarity compared to the comment?

10 odpovedi



Additional remarks (if any):

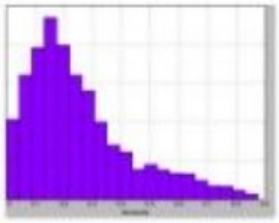
2 odpovede

Bug reports are always kind of more negative in nature, imo

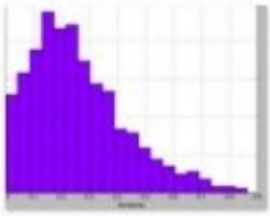
Overall discussion of a problem will lead to group knowledge being shared, whereas the initial comment is just the knowledge of one person. There is a higher likelihood that any individual comment will diverge from any other comment about a particular topic, but the average evens out because all information is shared over time.

STACK OVERFLOW: There are 4 histograms total. Each shows text similarity

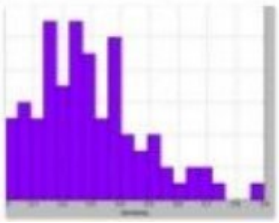
Histogram 1



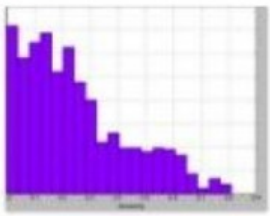
Histogram 3



Histogram 2

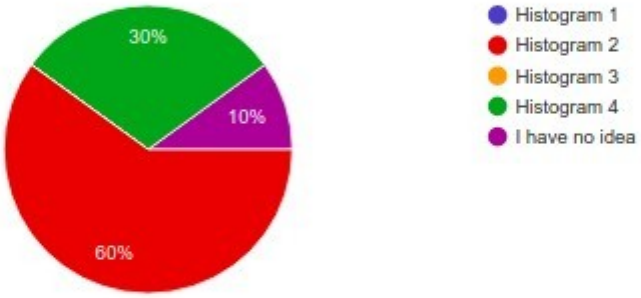


Histogram 4



STACK OVERFLOW: There are 4 histograms total. Each shows text similarity distribution between 1000 Git issues and StackOverflow discussions (similarity increases from left to right). 3 histograms show similarity for one specific project and similarities have been calculated for related but also unrelated Git-Stack item pairs. One histogram though contains just the similarities of SO discussions and their matching Git issues which they talk about. Can you identify that histogram?

10 odpovedi



Additional remarks (if any):

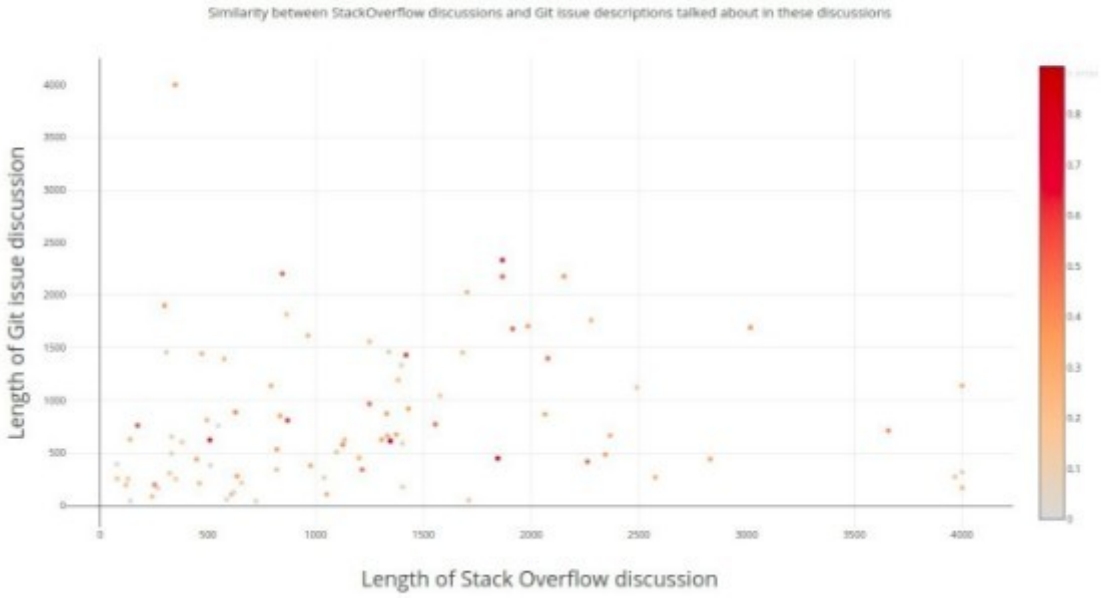
3 odpovede

This question is really opaque and hard to understand.

I have no idea what these are measuring.

I feel like comparisons across the entire set would even out more often, so I'm going with the irregular graph.

STACK OVERFLOW: The following scatter plot shows three dimensions - length



STACK OVERFLOW: The following scatter plot shows three dimensions - length of StackOverflow question regarding a git issue, length of Git description of that very same issue, similarity of these two texts. Each dot represents an issue and it's color represents the similarity of issue description and Stack Overflow discussion. Explain whether you see a relationship between the three data series or not! Interactive 3D Scatter plot can be found here -> https://plot.ly/~DurkoMatko/3

6 odpovedi

The shorter the SO discussion, the more likely it is clearly an issue recorded on git(hub - I assume)... maybe?
Not really, SO/GitHub are basically a preferred communication styles.
Seems like longer descriptions are more likely to be similar, maybe because they are copy pasted? I would check for a bias in the underlying similarity finder though
it looks like the longer both things are, the more similar they are. though... what are examples of this? why would git issues also be discussed on stack overflow? I need details!
I don't see any relationship
Seems like there's a pretty high correlation between similarity of short length github issue discussions and longer SO discussions. Could be that bugs are reported on SO first, then discussed, then re-reported and opened on GitHub once the underlying cause is found.

LAST question: Which project have you lately contributed to?

- ○ AngularJS
- ○ NodeJS
- ○ EmberJS
- ○ VueJS
- ○ CakePhp
- ○ Laravel

LAST question: Which project have you lately contributed to?

10 odpovedi