Language Communities on GitHub

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

May 1, 2017

How do different language communities communicate?

Important to understand in order to…

- **Improve contributions**
  - Helps new language designers cultivate communities
- Better understand different customs
  - **Enables newcomers** to join the community (know what to expect)

Similar studies in the past found...

- Contributors and repository owners use interactions to **evaluate each other**
- **History** with the project increased probability of whether a contribution is accepted
- Newcomers face hurdles integrating into an online community

Instead: focus on interactions within community

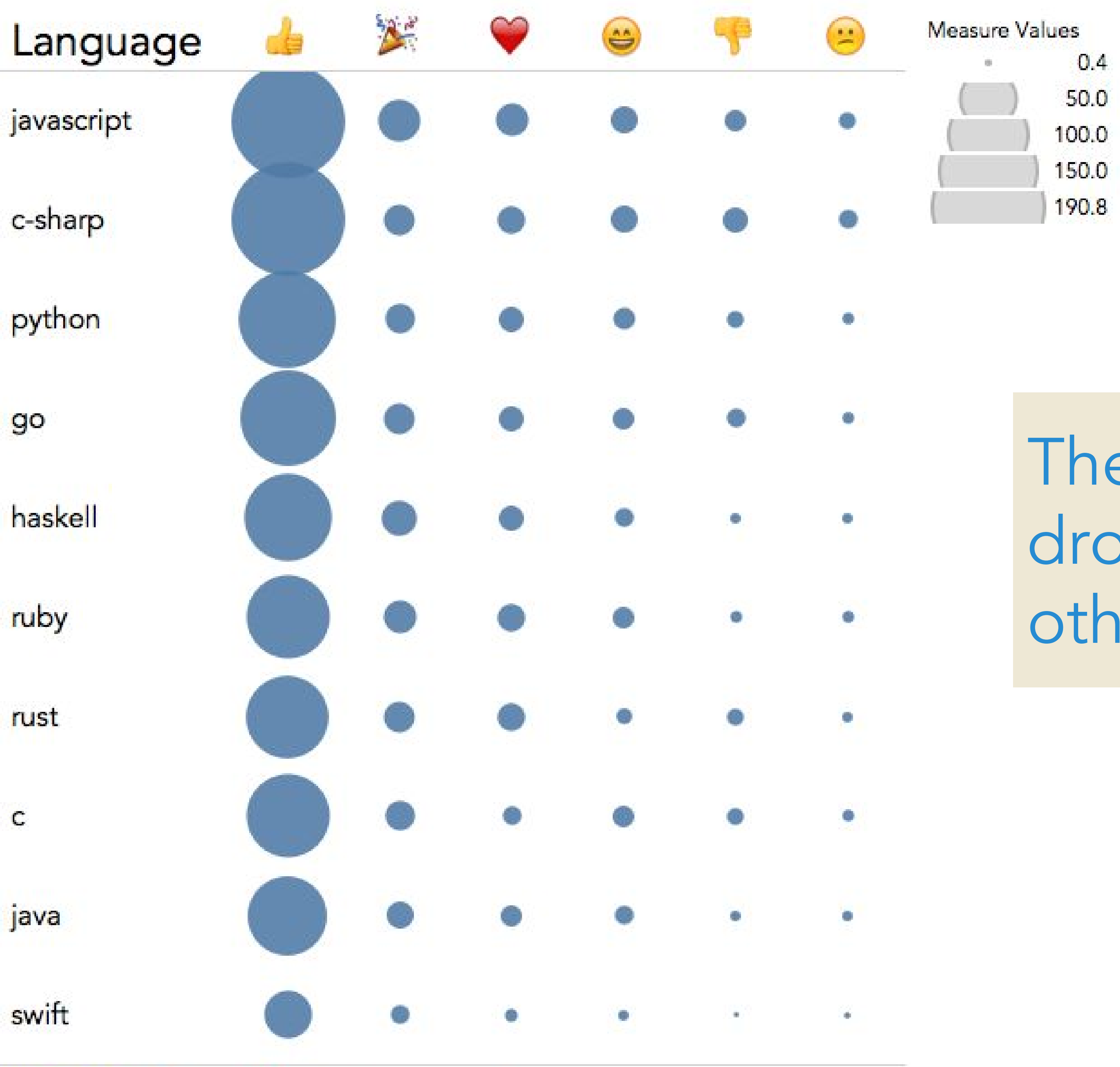
- Focus on interpersonal interactions
  - Sentiment: positive vs negative
  - Topic: what are they talking about?
- Jargon & trending topics in each community
  - Helps newcomers
- How one language community relates to others
  - Helps contributors moving from one community to another

Sampled data from GitHub API

- Randomly sample public projects in **each language**
  - medium-sized (between 1,000 – 4,000 stars)
  - not abandoned (updated this year)
- GitHub Issues API gives us:
  - **reactions** data for each comment
  - **body text** for each issue description

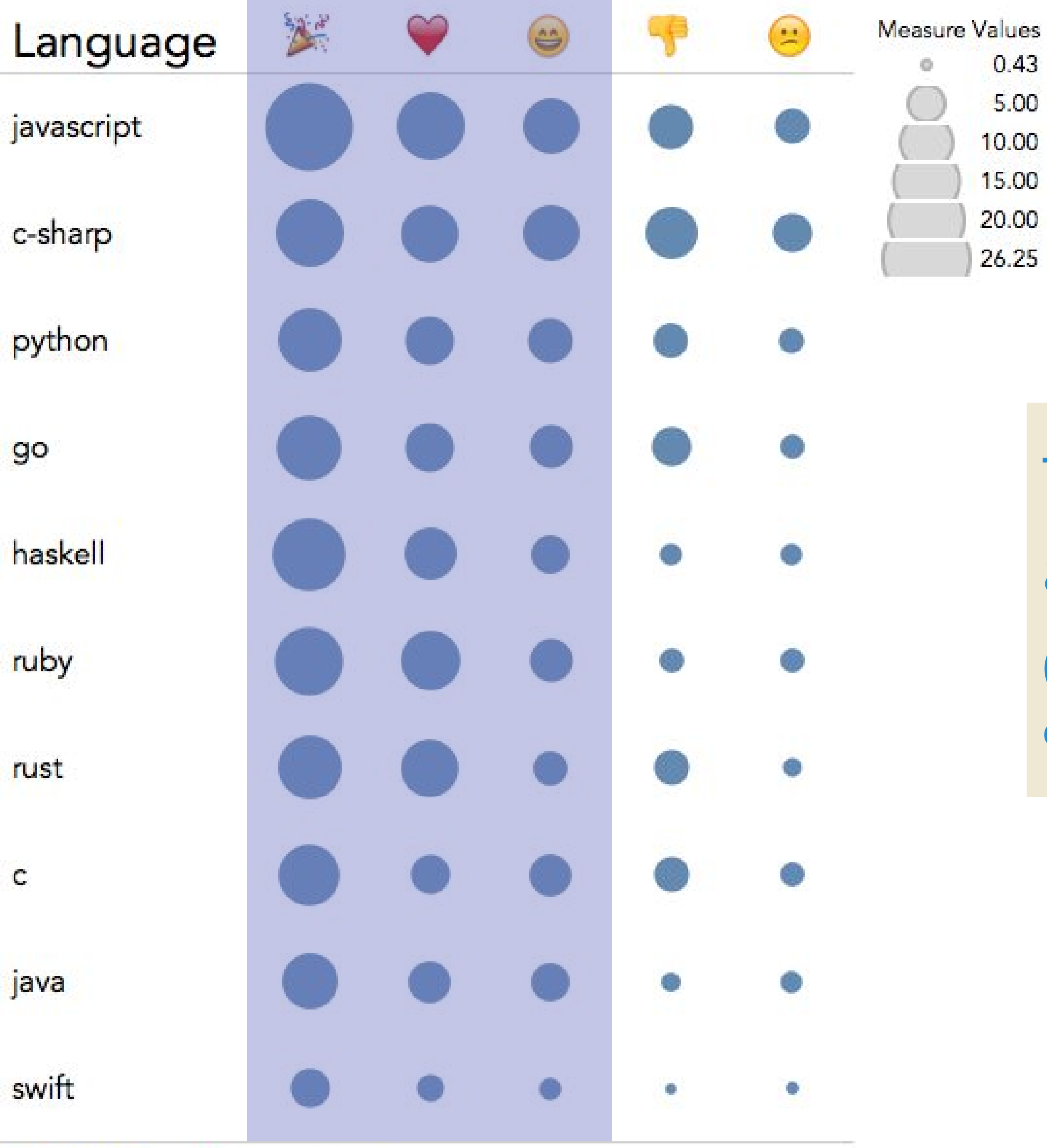
Emoji reactions reflect interpersonal actions

- One “reaction” involves at least **two people** (usually more)
  - commenter + reacter(s)
- Emoji capture interpersonal **emotions**
  - Simple metric
  - Captures sentiment, quantity, etc.



👍, 👎, 😄, 😕, 🎉 and ❤️ (size) broken down by Language.

The 👍 emoji drowns out all other emoji

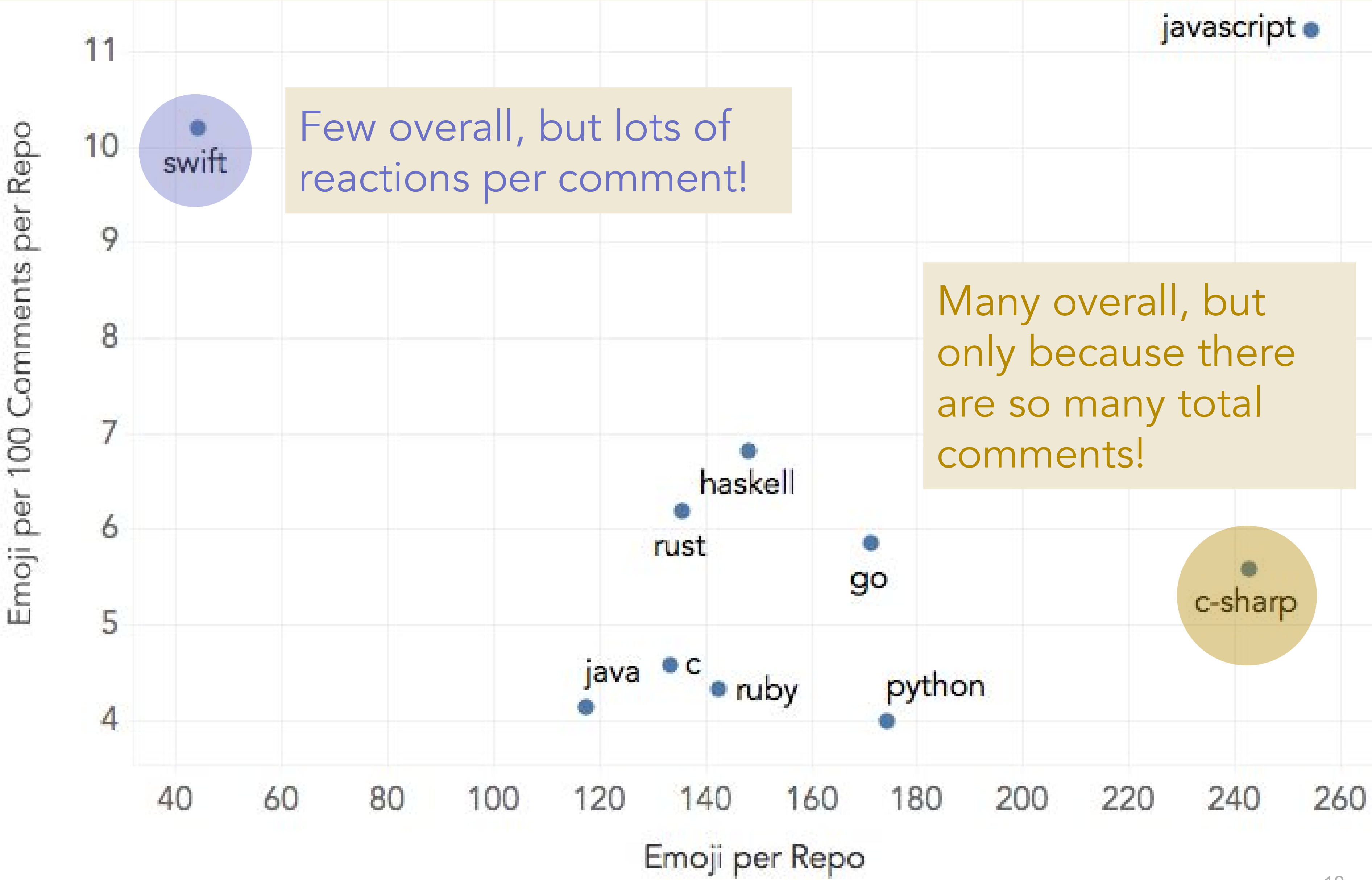


👎, 😀, 😕, 🎉 and ❤️ (size) broken down by Language.

The next top 3 are all positive

(passionate for their communities!)

Density per comment vs Total reaction quantity

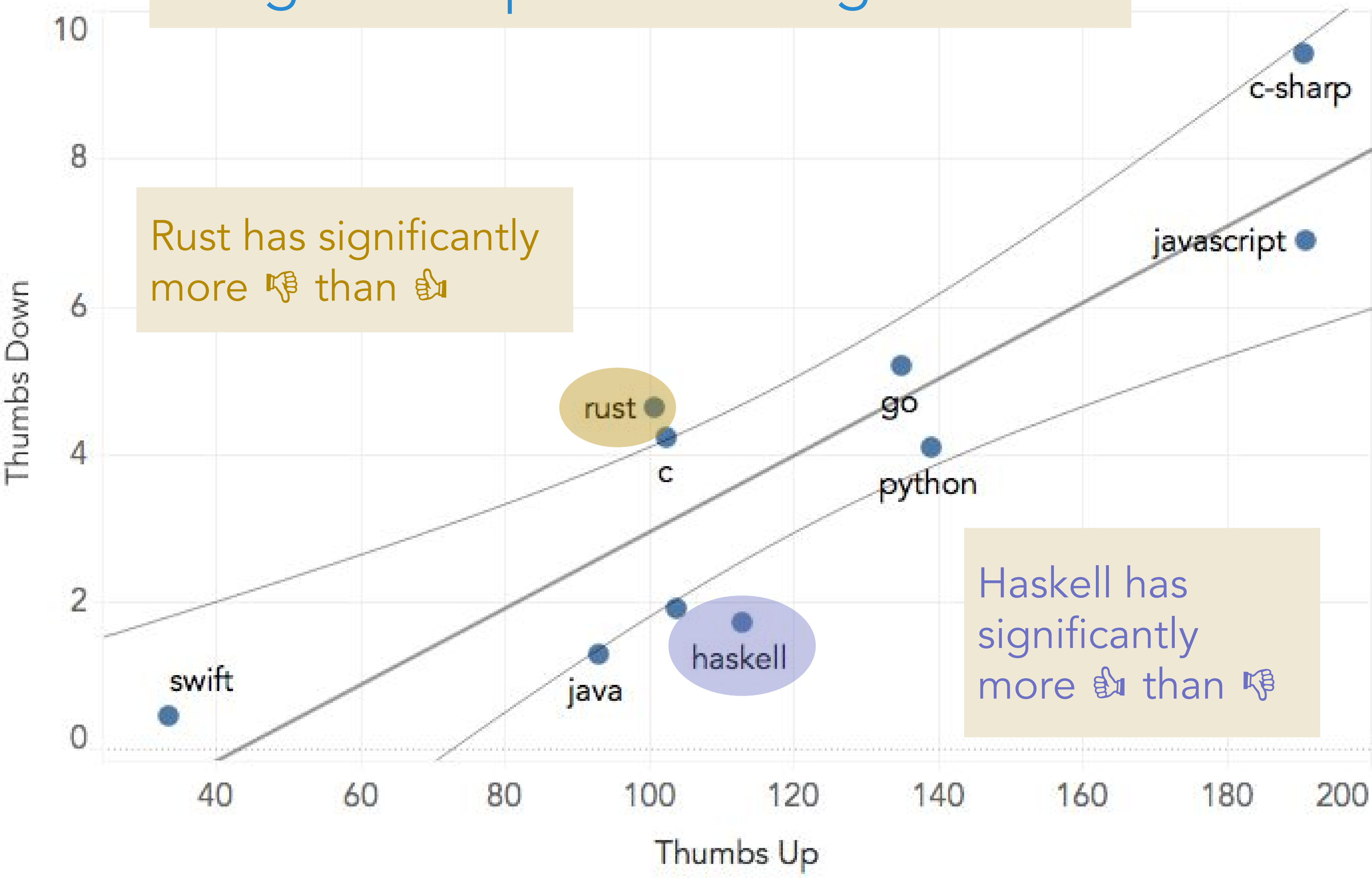


Few overall, but lots of reactions per comment!

Many overall, but only because there are so many total comments!

Avg(Total Count) vs. Emoji per 100 Comments. The marks are labeled by Language.

Usage of 👍 predicts usage of 👎



Rust has significantly more 👎 than 👍

Haskell has significantly more 👍 than 👎

Avg(Thumbsup) vs. Avg(Thumbsdown). The marks are labeled by Language.

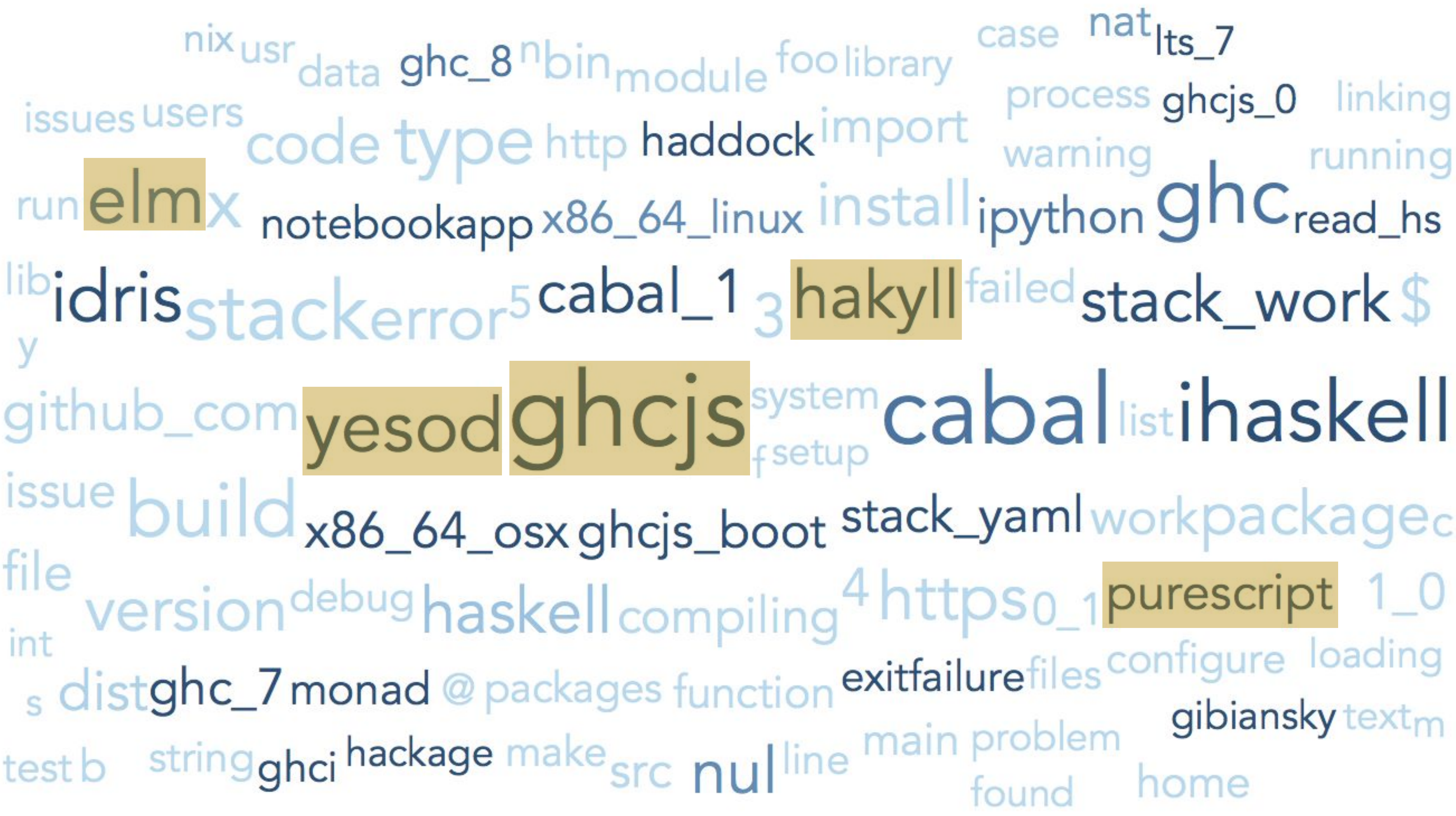
Emoji Reactions: Key Takeaways

- Overwhelmingly used to convey **positive emotion**
- Certain communities tend to be more positive overall
  - **Haskell**: super positive
  - **Rust**: more critical or negative
- Communities like Swift and JavaScript use reactions **abundantly**

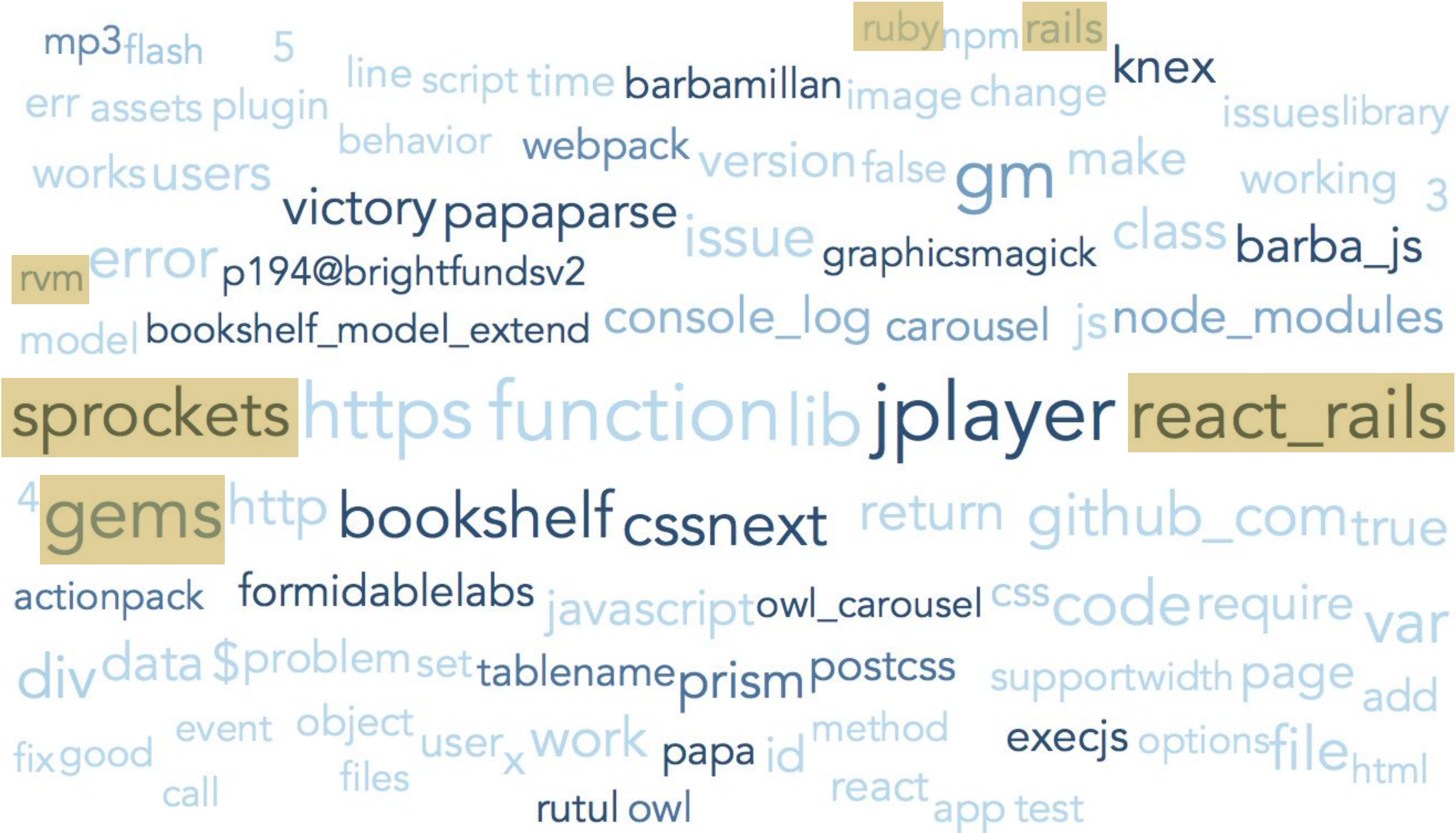
Unigram models expose topical trends

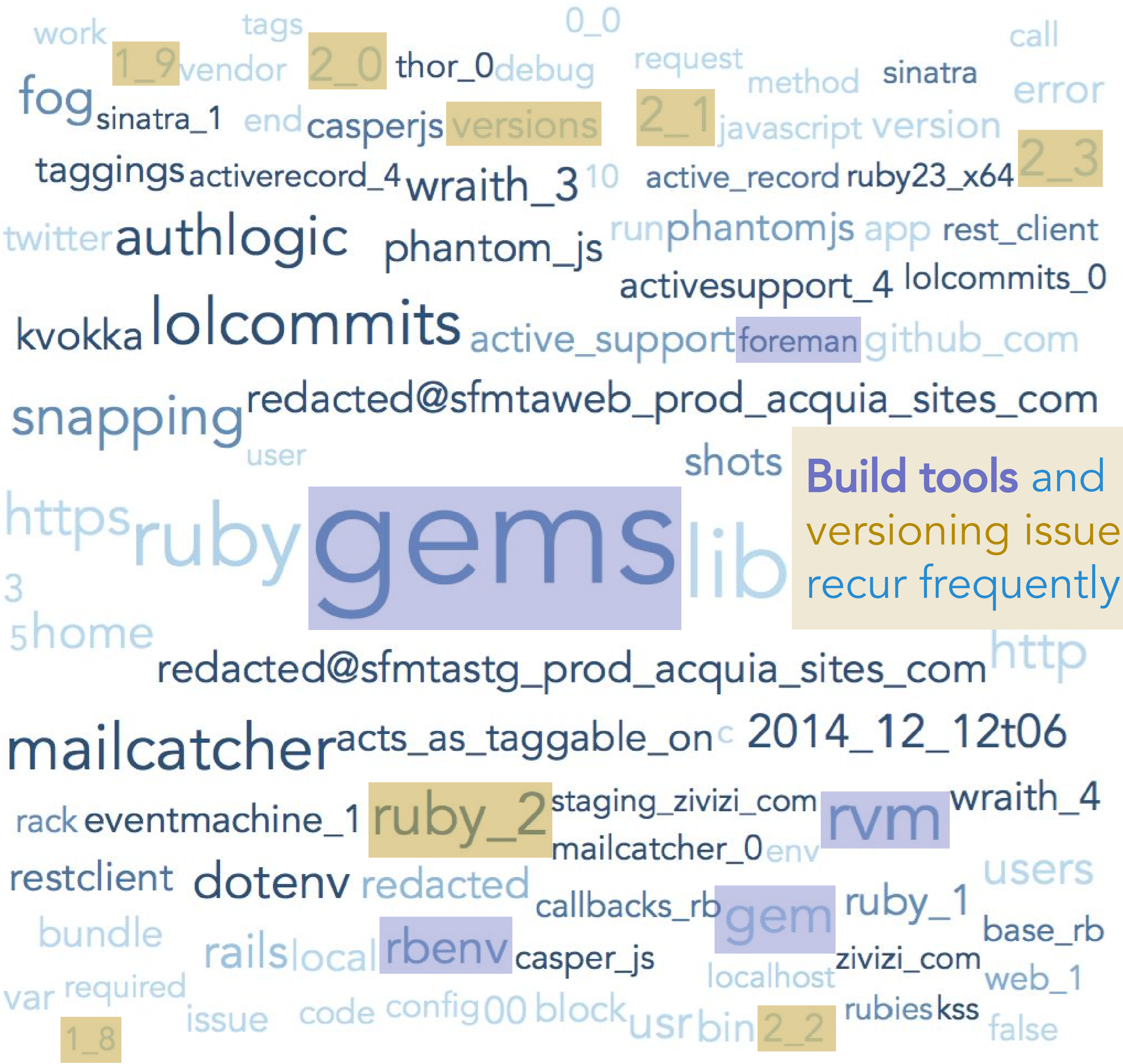
- Unigram model counts occurrences of words
  - also called “bag of words”
- Can tell us:
  - topics **common** to all languages
  - topics **unique** to specific languages
- Technical note for the curious
  - uses **tf-idf** scoring under the hood

Lots of Haskell project discussing web tech



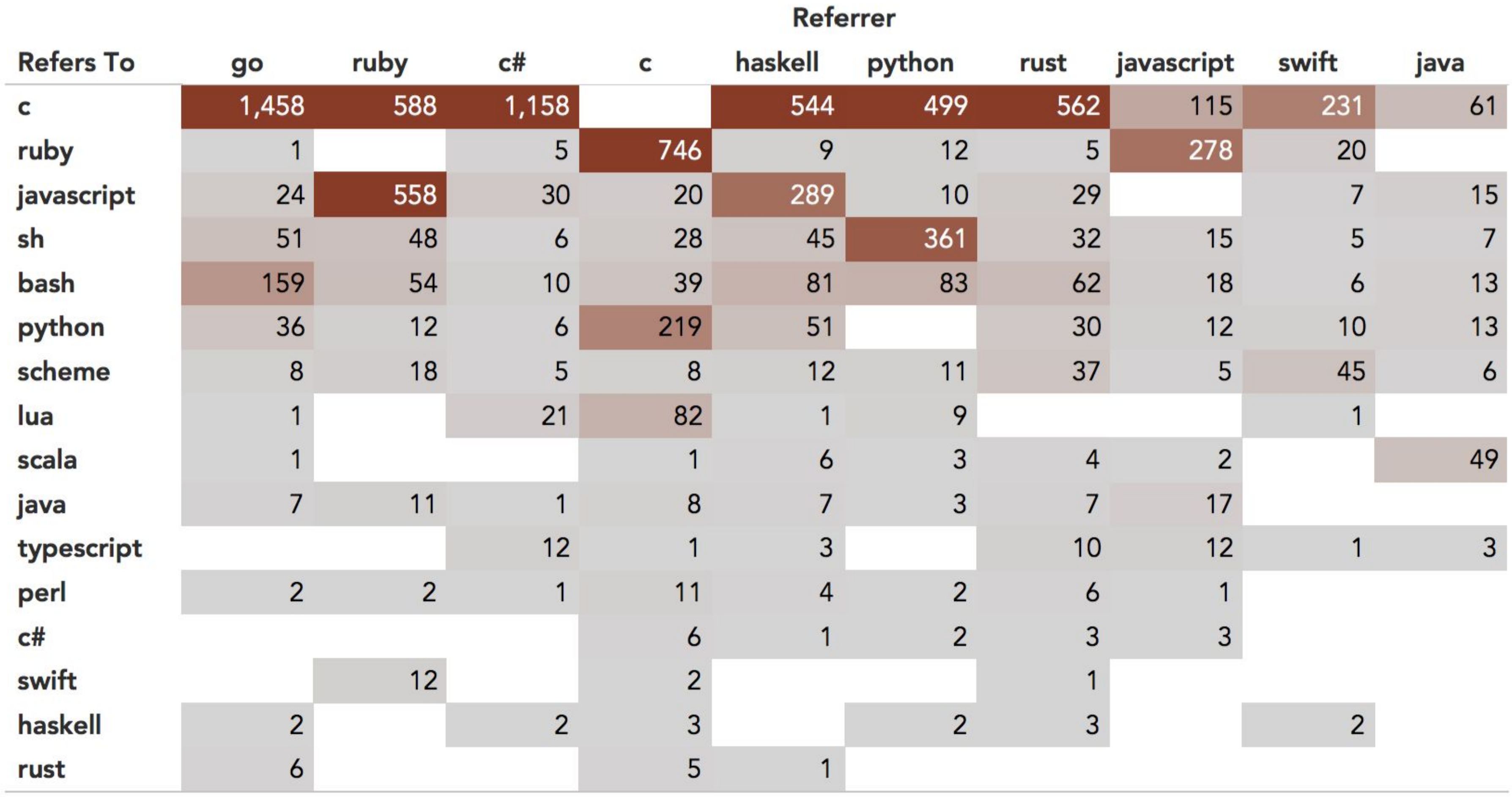
More crossover for web development: Ruby + JavaScript




work tags 0_0 call
1_9 vendor 2_0 thor_0 debug request method sinatra error
fog sinatra_1 end casperjs versions 2_1 javascript version 2_3
taggings activerecord_4 wraith_3 10 active_record ruby23_x64
twitter authlogic phantom_js run phantomjs app rest_client
activesupport_4 lolcommits_0
kvokka lolcommits active_support foreman github_com
snapping redacted@sfmtaweb_prod_acquia_sites_com
user shots
https ruby gems lib
3
5 home
redacted@sfmtastg_prod_acquia_sites_com http
mailcatcher acts_as_taggable_on c 2014_12_12t06
rack eventmachine_1 ruby_2 staging_zivizi_com mailcatcher_0 env rvm wraith_4
restclient dotenv redacted callbacks_rb gem ruby_1 users
bundle base_rb
rails local rbenv casper_js localhost zivizi_com web_1
var required issue code config 00 block usr bin 2_2 rubies kss false
1_8

Build tools and
versioning issues
recur frequently (ex Ruby)

C, Ruby, and JavaScript are referred to often



	Referrer									
Refers To	**go**	**ruby**	**c#**	**c**	**haskell**	**python**	**rust**	**javascript**	**swift**	**java**
c	1,458	588	1,158		544	499	562	115	231	61
ruby	1		5	746	9	12	5	278	20	
javascript	24	558	30	20	289	10	29		7	15
sh	51	48	6	28	45	361	32	15	5	7
bash	159	54	10	39	81	83	62	18	6	13
python	36	12	6	219	51		30	12	10	13
scheme	8	18	5	8	12	11	37	5	45	6
lua	1		21	82	1	9			1	
scala	1			1	6	3	4	2		49
java	7	11	1	8	7	3	7	17		
typescript			12	1	3		10	12	1	3
perl	2	2	1	11	4	2	6	1		
c#				6	1	2	3	3		
swift		12		2			1			
haskell	2		2	3		2	3		2	
rust	6			5	1					

Count as an attribute (color) broken down by Referrer vs. Refers To. The view is filtered on Refers To, which keeps 16 of 36 members.

Text Analysis: Key Takeaways

- Certain **language stereotypes** aren’t well founded
  - Haskell: used in web development, not just compiler development
- **C** and **JavaScript** permeate many languages
  - Beginners will have to know these in addition to the specific language of the community
- **Web technologies** cut across language boundaries
  - Might want to refocus on “web development” instead of “language”
- **Building and versioning** is common to all communities
  - How can we onboard beginners to this community’s build tooling?

Questions?