Mining GitHub

Anuraag, Apoorva, Avani, Jui, Vincent

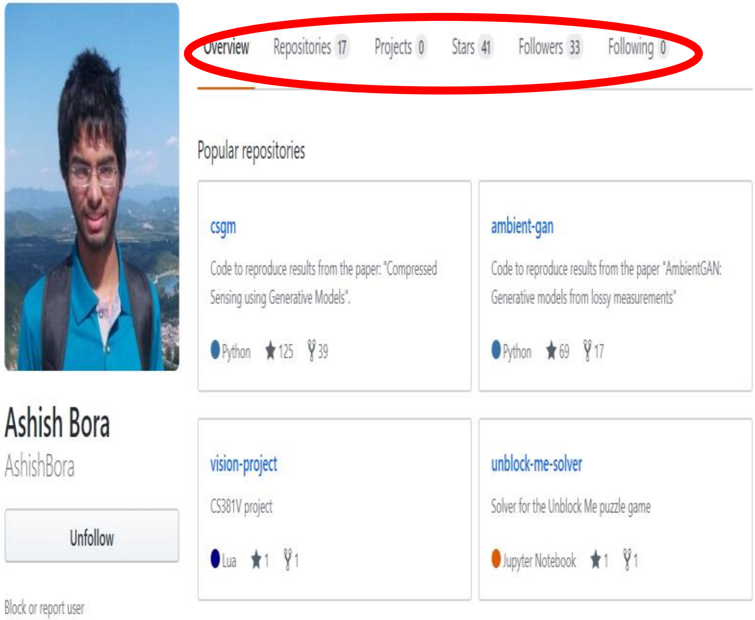
Goal:
To analyse Interest Graph Networks on GitHub

Possible Applications

- To extend this functionality to design a product for hiring cohesive teams
- Creating communities of top-notch researchers or developers in the domain
- To build ‘recommendation engine’ that GitHub currently lacks

Approach

Not-so-popular user



Overview Repositories 17 Projects 0 Stars 41 Followers 33 Following 0

Popular repositories

csgm

Code to reproduce results from the paper: "Compressed Sensing using Generative Models".

● Python ★ 125 ⑂ 39

ambient-gan

Code to reproduce results from the paper "AmbientGAN: Generative models from lossy measurements"

● Python ★ 69 ⑂ 17

vision-project

CS381V project

● Lua ★ 1 ⑂ 1

unblock-me-solver

Solver for the Unblock Me puzzle game

● Jupyter Notebook ★ 1 ⑂ 1

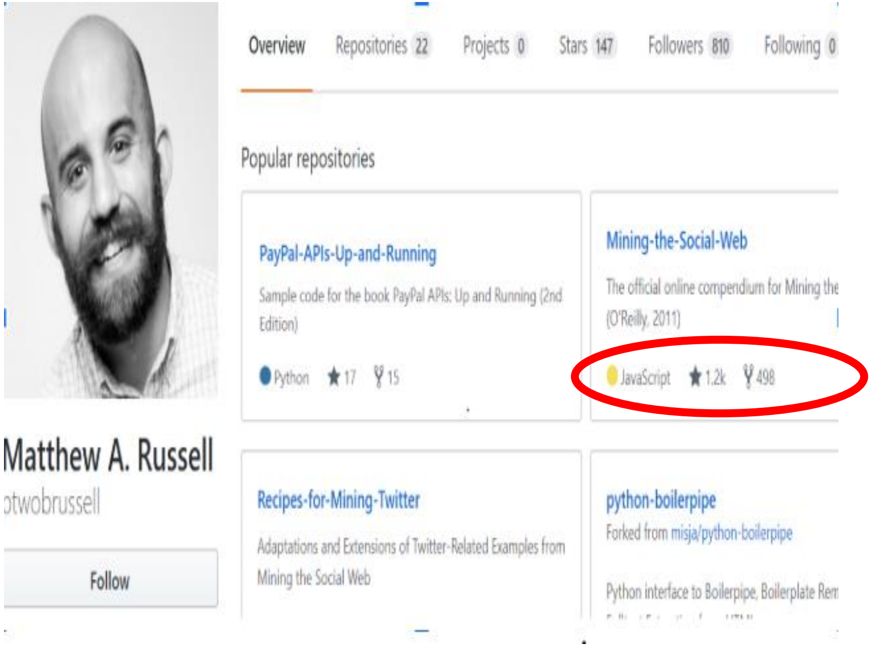
Ashish Bora

AshishBora

Unfollow

Block or report user

Popular repository



Overview Repositories 22 Projects 0 Stars 147 Followers 810 Following 0

Popular repositories

PayPal-APIs-Up-and-Running

Sample code for the book PayPal APIs: Up and Running (2nd Edition)

● Python ★ 17 ⑂ 15

Mining-the-Social-Web

The official online compendium for Mining the (O'Reilly, 2011)

● JavaScript ★ 1.2k ⑂ 498

Recipes-for-Mining-Twitter

Adaptations and Extensions of Twitter-Related Examples from Mining the Social Web

python-boilerpipe

Forked from misja/python-boilerpipe

Python interface to Boilerpipe, Boilerplate Rem

Matthew A. Russell

ptwobrussell

Follow

Methodology

- Query GitHub’s API
- Model the data
- Discovering the stargazers of our popular repository
- Exploring the graphical structures
- Use other APIs to model social connections among stargazers to understand their common interests

Our tool-kit

Packages & Libraries used

PyGithub

- List Stargazers API
- List of repositories starred API
- User Follower API

NetworkX

Nuances taken care of

1) **Exploring the API**

- Unauthenticated token - 60 unauthenticated requests per hour
- Authenticated token - 10000 requests per hour rate limit

2) **Avoidance of naming collisions**

- Username with same repository name
- Different users having repositories with same name (appended the suffix to the add_node())

Network Analysis on Ashish Bora’s github profile

Explaining levels

1) **Level 1**

- Start from Ashish Bora’s github profile with 32 followers

    2) **Level 2**

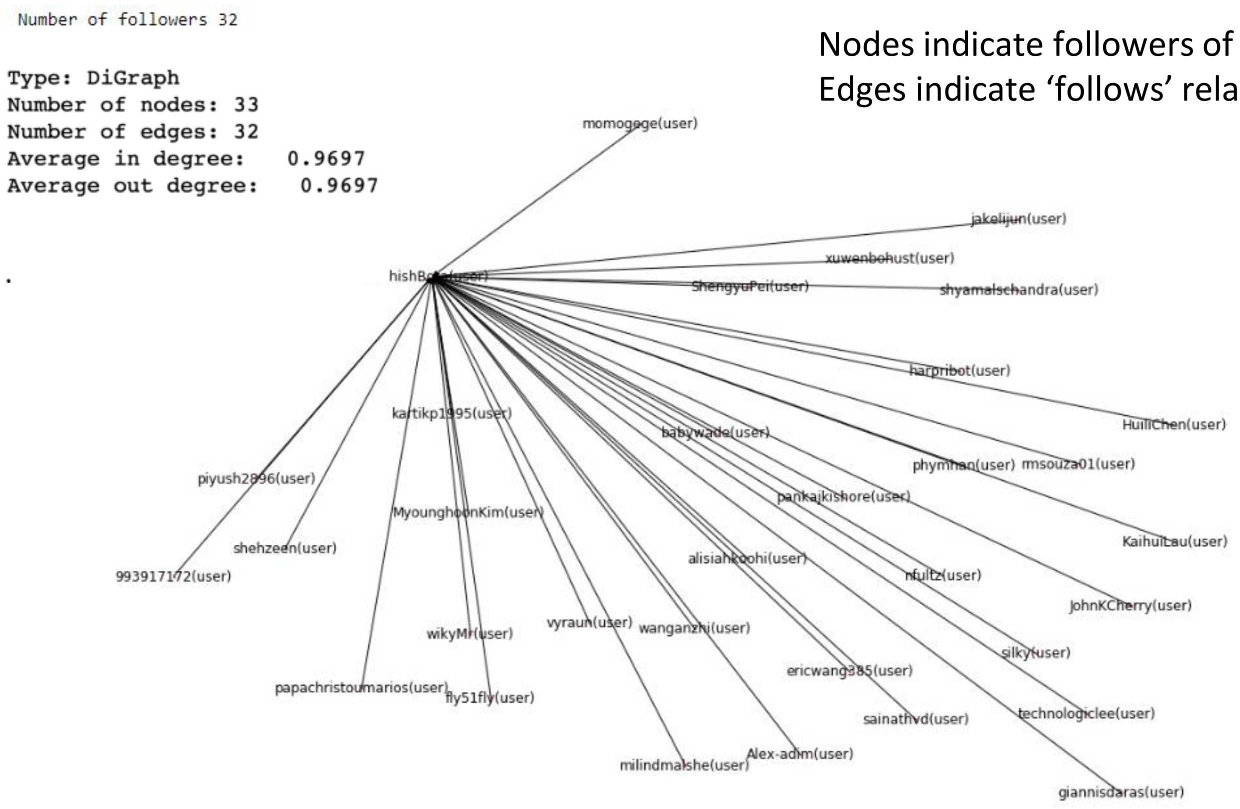
    - Ashish Bora’s followers’ followers

        3) **Level 3**

        - Ashish Bora’s followers’ followers’ followers

E G O G R A P H

Nodes indicate followers of Ashish
Edges indicate ‘follows’ relationship

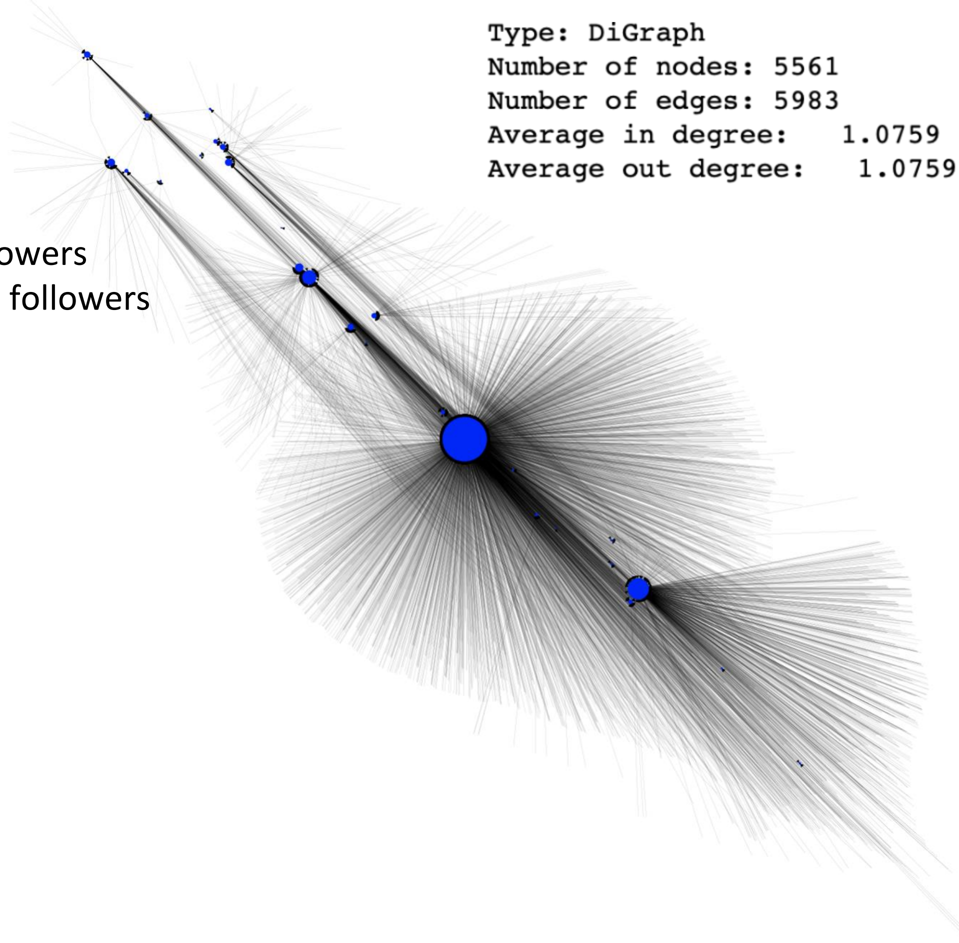


```
Number of followers 32

Type: DiGraph
Number of nodes: 33
Number of edges: 32
Average in degree:   0.9697
Average out degree:   0.9697
```

Level 2 Graph

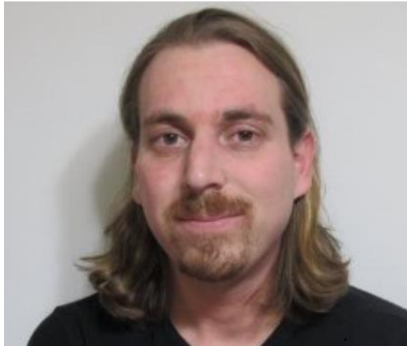
Nodes - Ashish + 32 Followers + Followers of 32 followers
Edges - Relationship from followers' followers to 32 followers
and among Ashish 32 followers

Neil Fultz - 3.7 K followers
Fly51fly - 811 followers
Silky - 350 followers
Tehcnologiclee - 399 followers



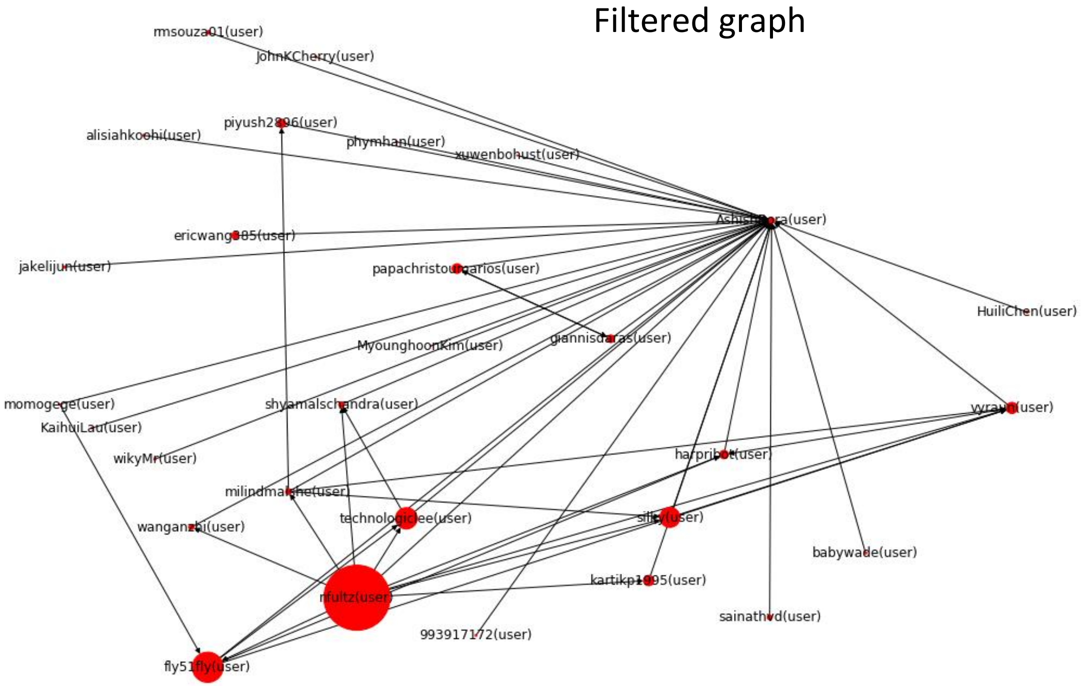
```
Type: DiGraph
Number of nodes: 5561
Number of edges: 5983
Average in degree:   1.0759
Average out degree:   1.0759
```

Who’s famous among his followers?



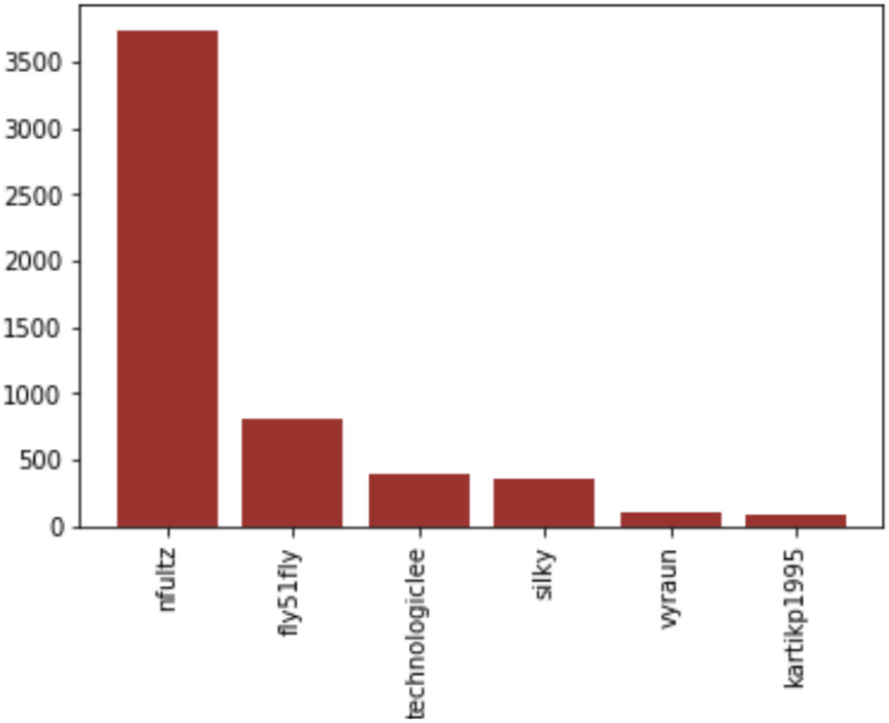
Neal Fultz

- 3.7K GitHub followers
- 3.4K Stars
- Principal Data Scientist
- Author of several open-source R and Python packages for Bayesian inference and optimization

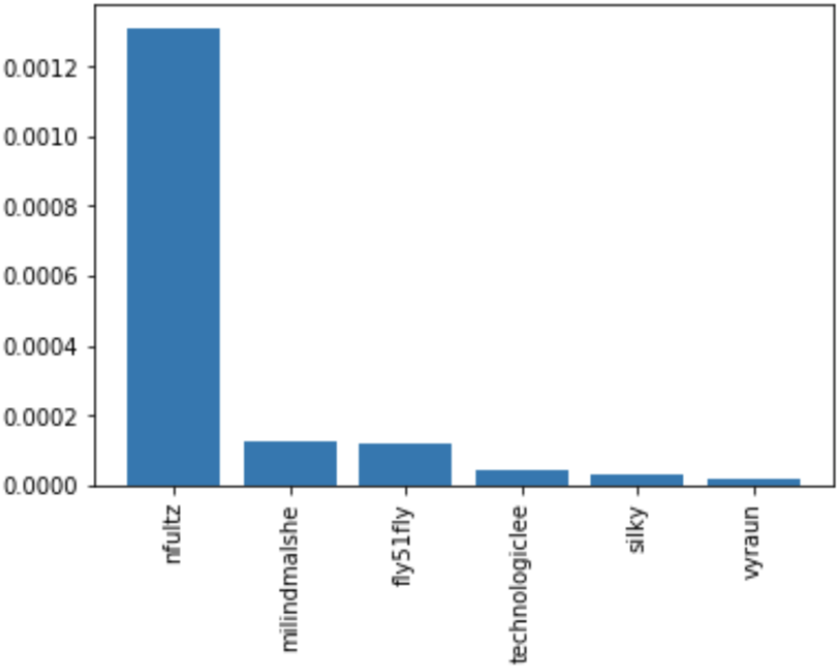


Filtered graph

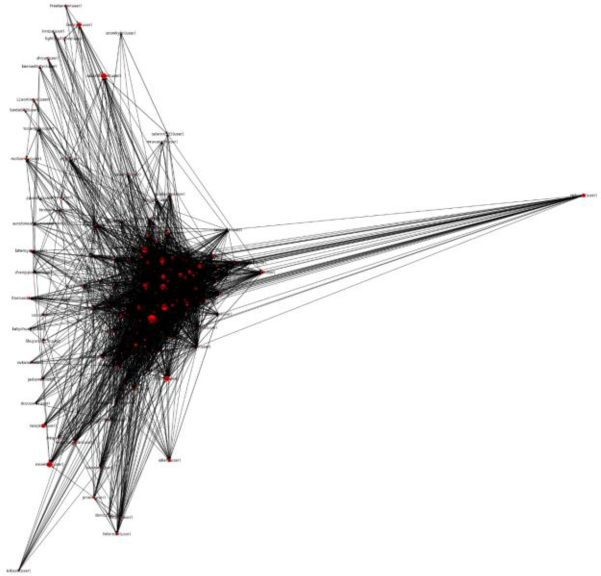
In-degree



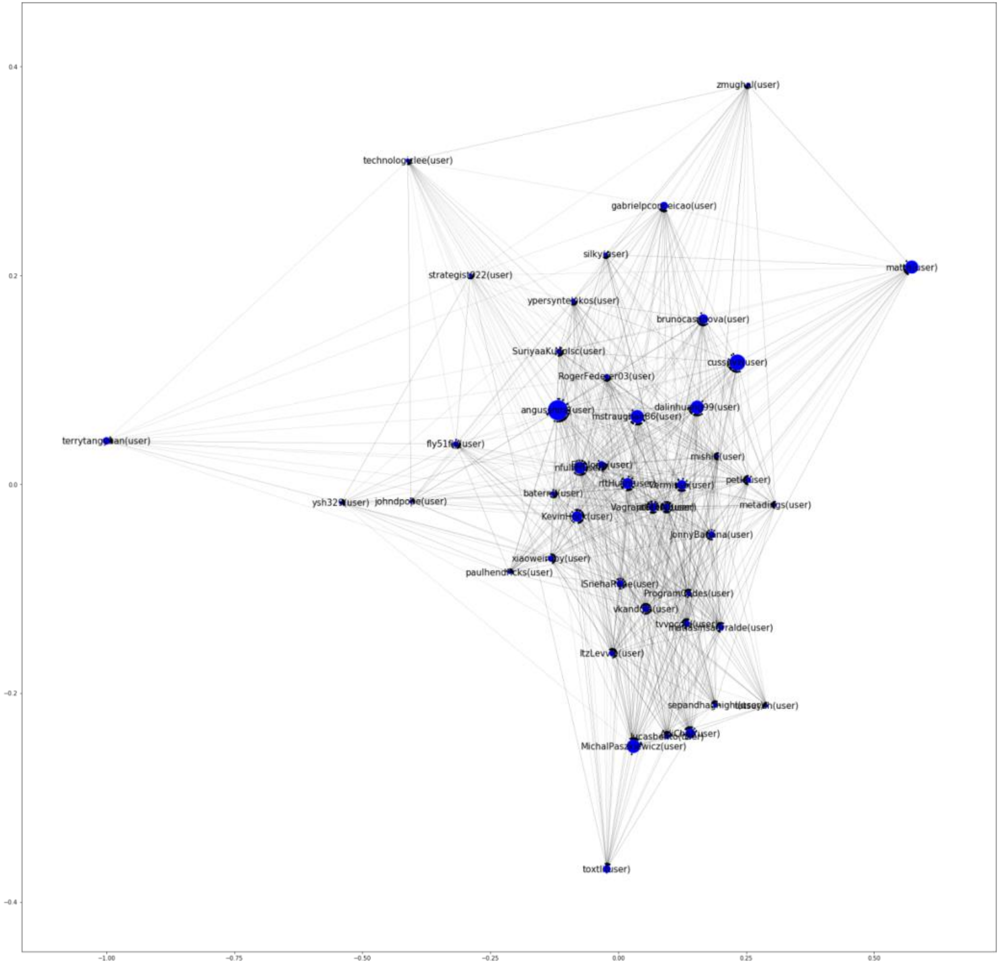
Betweenness



Level 3 Graph



```
Type: DiGraph
Number of nodes: 100916
Number of edges: 192206
Average in degree:   1.9046
Average out degree:   1.9046
```



User Network Analysis

Number of followers (level 1) = 32

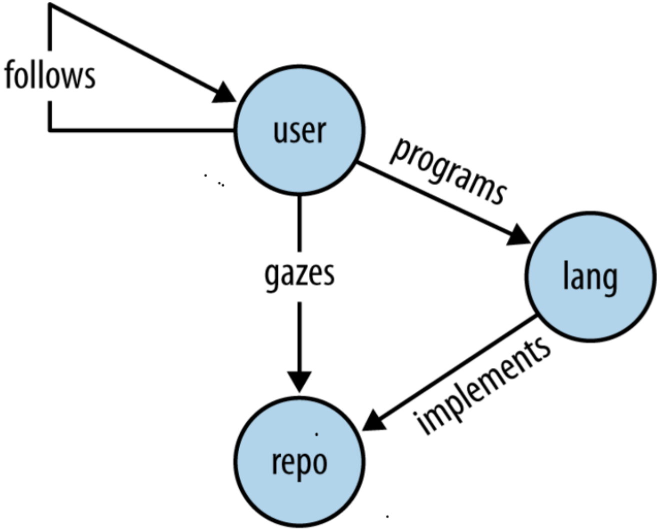
Levels	Nodes	Edges	In degree	Out Degree
Level 1	33	32	0.9697	0.9697
Level 2	5552	5974	1.0760	1.0760
Level 3	100916	192206	1.9046	1.9046

Network Analysis on ‘Mining the social web’ repo by Matthew Russell (ptwobrussell)

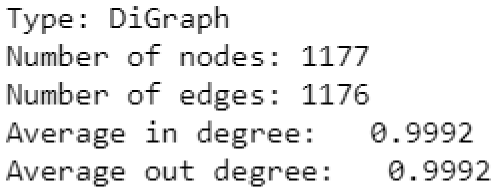
Repo Network Analysis

Number of Stargazers = 1176 (nodes 1177)

Levels	Nodes	Edges	In degree	Out Degree
Level 1	1177	1176	0.9992	0.9992
Level 2	1177	2811	2.3883	2.3883
Level 3	125175	292289	2.3350	2.3350

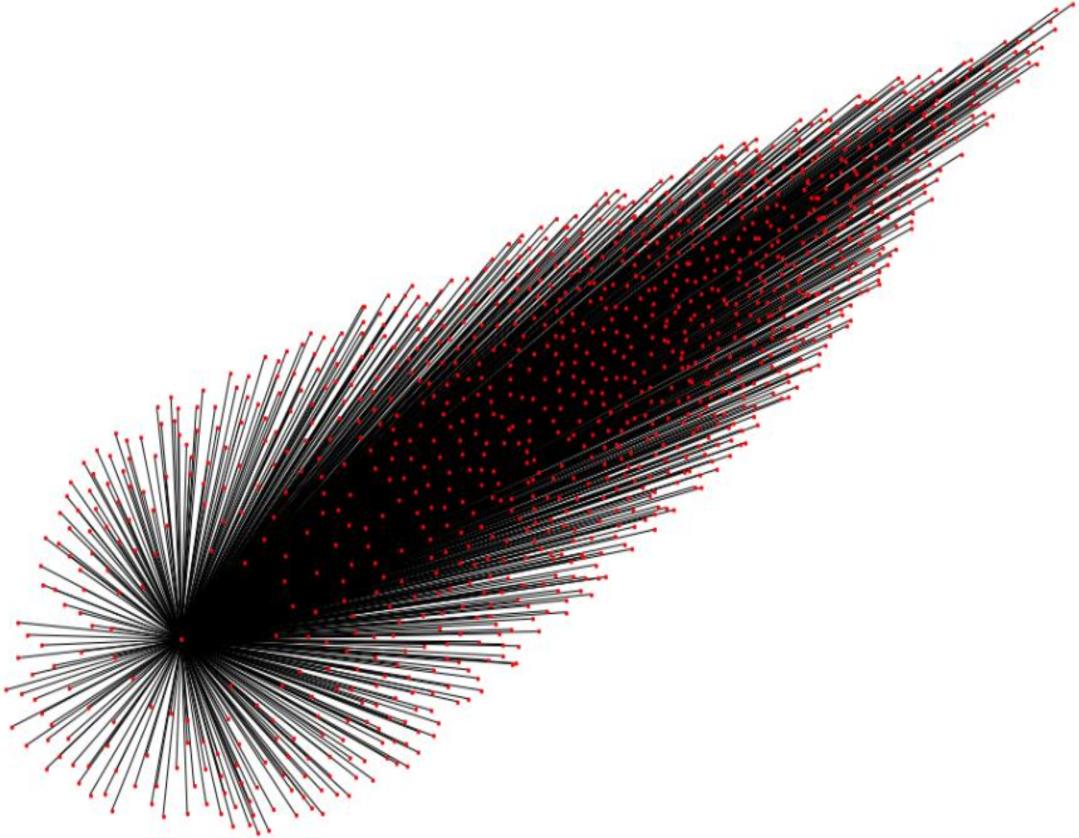


Ego graph of repo - Level 1



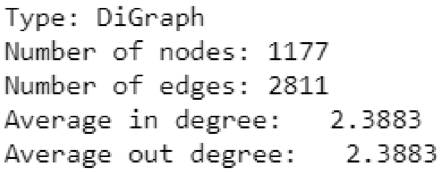
```
Type: DiGraph
Number of nodes: 1177
Number of edges: 1176
Average in degree:   0.9992
Average out degree:   0.9992
```

Edges indicate ‘gazing’



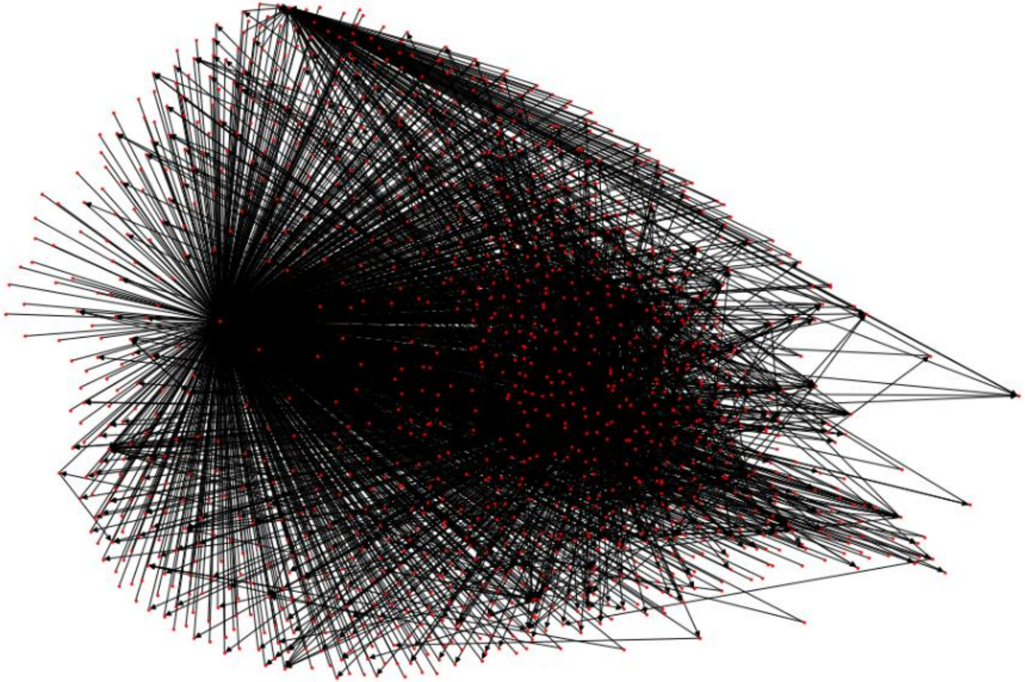
Level - 2
Connections among stargazers

In-degree and Out-degree increases



```
Type: DiGraph
Number of nodes: 1177
Number of edges: 2811
Average in degree:   2.3883
Average out degree:   2.3883
```

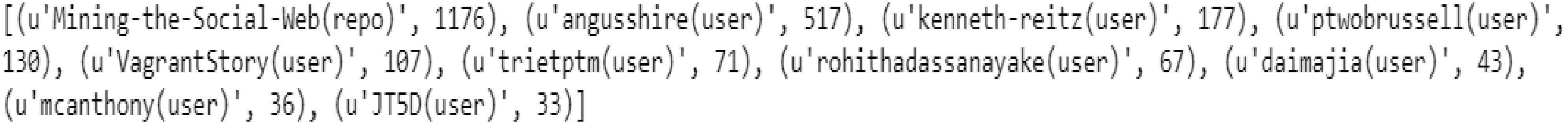
Edges indicate ‘repo gazing’ and ‘within gazers following’



Level 2 - Observations

Most popular users on the basis of **degree & followers among stargazers:**

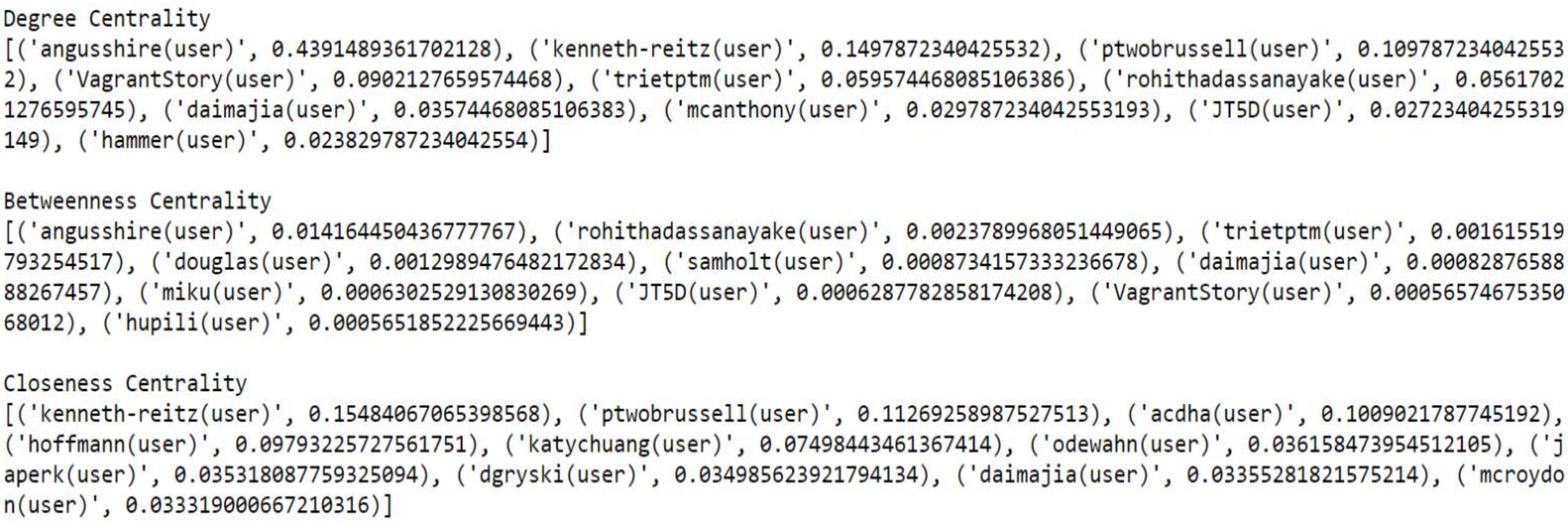
- **Angus Hung** - PhD UC Berkeley, 11.k followers & 213k stars
- **Kenneth Reitz** - Writer of @requests python library
- **Mathew Russell -** Owner & Author of the ‘Mining the social web’ repo
- **Minh Triet Pham Tran -** Forensics analyst with 864 followers & 17k stars
- **Daimajia -** Student, 22.1k followers and 2.9k stars



```
[(u'Mining-the-Social-Web(repo)', 1176), (u'angusshire(user)', 517), (u'kenneth-reitz(user)', 177), (u'ptwobrussell(user)',
130), (u'VagrantStory(user)', 107), (u'trietptm(user)', 71), (u'rohithadassanayake(user)', 67), (u'daimajia(user)', 43),
(u'mcanthony(user)', 36), (u'JT5D(user)', 33)]
```

Level 2 - Centrality analysis

Dropped the seed node to highlight the users acting as bridges



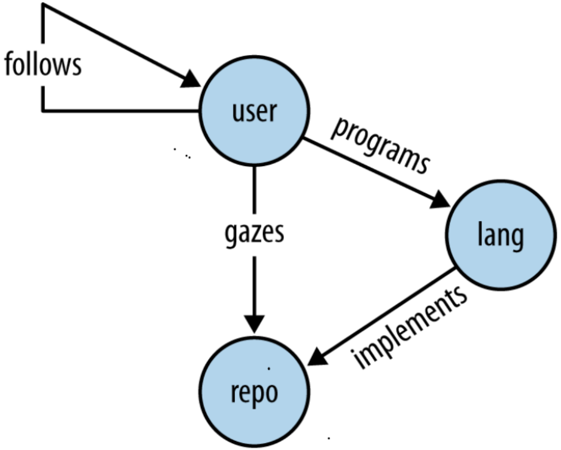
```
Degree Centrality
[('angusshire(user)', 0.4391489361702128), ('kenneth-reitz(user)', 0.1497872340425532), ('ptwobrussell(user)', 0.109787234042553
2), ('VagrantStory(user)', 0.0902127659574468), ('trietptm(user)', 0.059574468085106386), ('rohithadassanayake(user)', 0.0561702
1276595745), ('daimajia(user)', 0.03574468085106383), ('mcanthony(user)', 0.029787234042553193), ('JT5D(user)', 0.02723404255319
149), ('hammer(user)', 0.023829787234042554)]

Betweenness Centrality
[('angusshire(user)', 0.014164450436777767), ('rohithadassanayake(user)', 0.0023789968051449065), ('trietptm(user)', 0.001615519
793254517), ('douglas(user)', 0.0012989476482172834), ('samholt(user)', 0.0008734157333236678), ('daimajia(user)', 0.00082876588
88267457), ('miku(user)', 0.0006302529130830269), ('JT5D(user)', 0.0006287782858174208), ('VagrantStory(user)', 0.00056574675350
68012), ('hupili(user)', 0.0005651852225669443)]

Closeness Centrality
[('kenneth-reitz(user)', 0.15484067065398568), ('ptwobrussell(user)', 0.11269258987527513), ('acdha(user)', 0.100921787745192),
('hoffmann(user)', 0.09793225727561751), ('katychuang(user)', 0.07498443461367414), ('odewahn(user)', 0.036158473954512105), ('j
aperk(user)', 0.035318087759325094), ('dgryski(user)', 0.034985623921794134), ('daimajia(user)', 0.0335281821575214), ('mcroydo
n(user)', 0.033319000667210316)]
```

Level 3
Adding starred repositories of the stargazers

Nodes - Seed Repository + 1176 Stargazers + Starred Repositories of stargazers
Edges - ‘Stargazing’

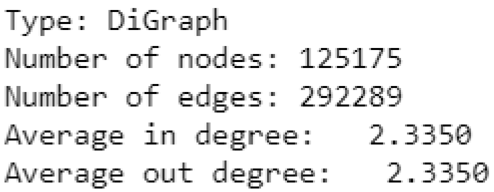
Repositories are engaging to this community

- **('Mining-the-Social-Web(repo)', 1176)**
- ('bootstrap(repo)', 213)
- ('tensorflow(repo)', 207)
- ('d3(repo)', 203)
- ('dotfiles(repo)', 186)
- ('free-programming-books(repo)', 158)
- ('models(repo)', 151)
- ('Mining-the-Social-Web-2nd-Edition(repo)', 144)



Level 4
Adding the language of the starred repositories

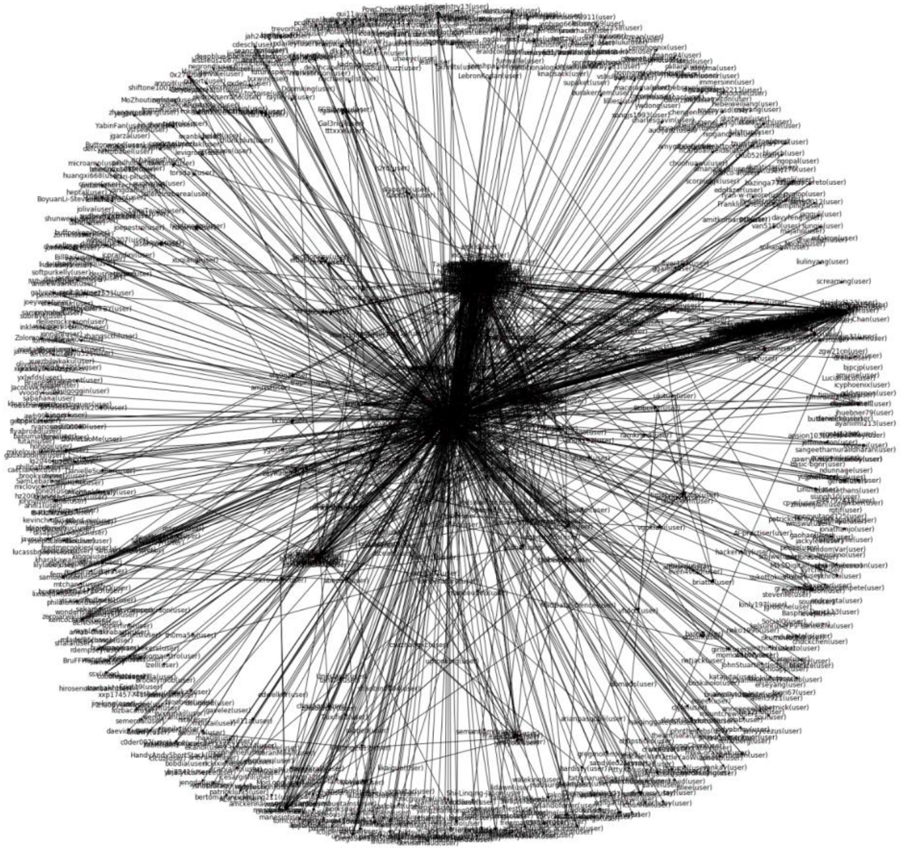
Nodes - Level 3 nodes + Languages used by the repository
Edges - Level 3 edges of stargazing + languages linked to both users and repositories



```
Type: DiGraph
Number of nodes: 125175
Number of edges: 292289
Average in degree:   2.3350
Average out degree:   2.3350
```

Level 4 - Observations

- Popular languages: Python and Javascript
- Number of Python programmers: 1083
- Number of JavaScript programmers: 1076
- Number of programmers who use JavaScript and Python: 1083
- Number of programmers who use JavaScript but not Python: 93



Detecting Communities by removing super nodes (Repository itself)

Moving forward

- Apart from centrality measures, we can use cliques or bipartite algorithms to derive insights

- Use similarity metrics for two arbitrary users on Github based on common starred repos, common programming languages etc.

Thank You!