Analyzing the Implicit Social Network from GitHub Activities

Ran Tavory, Ruzvidzo Ngulube, Jonathan del Campo, Brendan Danyluik

ABSTRACT

We develop a method of mining GitHub event activity to construct an implicit social network between GitHub users encompassing their GitHub repositories and the connections between them. We then utilize this network for interesting visualizations, including shortest path between users and finding the most influential users.

What are you trying to do?

Create an interactive UI with a graph that allows querying the activity of GitHub users and relations between GitHub users, in pticular, find the shortest path between two users (edges are defined by co-activity on the same project), Similar to the Erdős number in Mathematics [23], rate users in terms of their distance from some of GitHub "celebrity" user, for example distance from Linus Torvalds and Find the most influential users for a certain technology scope, e.g. D3.

Extracting implicit social graph structure has been suggested before, for example Lima et al. [16] create a followers and contributors graphs to calculate statistical measures such as the rich club coefficient [21]. Thung et al.[20] take a similar approach of mining GitHub's event history to build a graph and compute statistics for this graph, such as graph's connectivity, average shortest path and PageRank[19]

Identifying influential users or opinion leaders has its roots in sociology and in recent years has been implemented in different internet systems using, one interesting comparison between some of the methods is the focus of Liu et al. in [17] where China's social network "Sina" and a local cluster of 4k students in Shanghai University are used to compare the accuracy of three different opinion leaders discovery methods, PageRank, HITS[14] and Synthesized Centrality (invented by the authors).

Hu et al.[12] Through the usage of HITS generate graphs that shows the influence and relationship between GitHub repositories and users. They analyze how specific repositories influence the development of the code of other repositories, and study how repositories rank over time.

From a slightly different angel Batista et al.[3] study the correlation between the properties that measure the strength of software social coding collaboration on GitHub; they several ways to measuring collaboration are presented, for example the Preferential Attachment (PA) which assumes that the more edges a node has (a user or a project), the more likely it will get more edges.

Hu et al.[11] focus on a very specific workflow of GitHub, namely the Follow-Star-Fork workflow and construct a graph based on these activities. They implement several quantitative measures, specifically User-Rank (like PageRank), HITS, H-index[24], Betweenness centrality[6], Spearman rank correlation[25], and Borda Count voting[22] to measure the influence of a user.

Badashian and Stroulia[2] measure the influence of GitHub users and specifically define and quantify what does influence mean in GitHub and whether the measured influence remains siloed to specific domain of content (e.g. technology or programming language).

Badashian et al.[1] combine two software development focused websites, GitHub and StackOverflow, to study the influence and contribution across these two networks. After studying the characteristics of each separate network a combination of the networks is studied as the correlation between the same user's activity in one network to the other.

del Fresno García et al.[5] research Twitter to conduct Social Network Analysis (SNA) and identify Social Media Influencers (SMIs). They reveal the existence of three different SMI typologies: disseminator, engager and leader. Even though this study is conducted on Twitter, perhaps some of its methodology can be applied to GitHub.

How is it done today; what are the limits of current practice?

We have not seen such interactive interface for GH as of today. What we have seen is statistical analysis of GitHub's graph data ([16], [12] and [3]) but none of the resources we have surveyed allows for interactive discovery.

There are databases with raw GitHub data ([8], [9] and [7]) and there is an API for GitHub's data[**?**] but again, a tool that structures this data in a social network structure to allow interactive discovery we did not find.

We plan to create an interactive interface such as the one studied by Hansen et al.[10] where a user usability study of the network analysis tool NodeXL is conducted and a set of guidelines of do's an don'ts is presented. We do not intend to use NodeXL specifically, yet user study conclusions are useful.

What's new in your approach and why will it be successful?

It is a simple way to discover connection between software developers and perhaps discover software developer communities. This can be useful for businesses trying to reach out to certain communities and looking for thought leaders in those communities. It can also be useful to any GitHub user trying to assess her contributions within GitHubs' community.

Who cares?

Businesses may care about community discovery for marketing purposes. Individuals may care about personal branding and achievements.

What difference and impact will it make, and to measure them?

The project, if successful, would become a popular tool within developers and businesses looking to study the social graph implied by GitHub, in particular finding connections to other specific users and identifying communities and their thought leaders.

It has been claimed by Casalnuovo et al.[4] that there is evidence for socialization as a precursor to joining a project and so we believe that presenting the social graph would open doors to more cooperation between users.

We plan to measure that by posting references to this project on reddit, twitter, hackernews and measure their popularity and engagement.

What are the risks and payoffs?

There are several notable risks, in particular - data may be difficult to obtain at scale. Although there are multiple sources, from our research we know that no single source encompasses all the aspects we require so we will have to merge multiple sources. Merging successfully is risky as well as obtaining a complete view of the data at scale. We've seen reports of missing data, incorrect data, inconsistent data and obfuscated data (for privacy reasons)[13].

Data is large so there will be scalability concerns, including collection, preprocessing and serving. How to properly handle such large data and provide fast enough access for an interactive user experience may be challenging.

Payoffs: first, community engagement. Later - perhaps a business opportunity.

How much will it cost?

We will collect the data (from multiple sources), then process it (on a Spark) and store it for serving in a graph database (Neo4j[18]). Serving costs include the collector process (1-2servers), the spark cluster (3 nodes or more), at least one Node4j server and a web server. On top of that, networking (ingress and egress) and some storage costs. Back of the envelope, let us assume each server costs 40cent per hour (EC2 *m5.2xlarge*). Let us assume that for ongoing serving we need two servers (Neo4j and Web) for 2 months (60 days) and for the collection we need 4 servers (for the collection period, let's say 1 month), we roughly have $\$0.4 \times 24 \times (2 \times 60 + 4 \times 30) = \$2304$. This is not cheap and we will look for ways to get funding and reduce costs.

How long will it take?

We think data collection and processing can complete within one month and writing the interface on top of it will take another month but these can be parallelized.

If we have more time then another interesting aspect of the data we process is a Sentiment Analysis of Developers' Comments on GitHub, as suggested by Kumar et al.[15]. We could enhance our network analysis with an aggregation of the sentiment of the comments.

What are the midterm and final checks for success and how will progress be measured?

Table 1 summarizes the plan of work and checks

Table 1: Plan of Work

Work item	Who	Checks
Raw Data Collection	Ran Tavory	After 2 weeks a skin version, complete by 4 weeks
Data Augmentation	Ruzvidzo Ngulube	After 1 week works on stub data, complete by 4 weeks
UI Query and web server	Brendan Danyluik	After 2 week works on stub data, complete by 4 weeks
UI Presentation	Jonathan del Campo	After 2 week works on stub data, complete by 4 weeks

REFERENCES

[1] Ali Sajedi Badashian, Afsaneh Esteki, Ameneh Gholipour, Abram Hindle, and Eleni Stroulia. 2014. Involvement, Contribution and Influence in GitHub and Stack Overflow. In *Proceedings of 24th Annual International Conference on Computer Science and Software Engineering* (Markham, Ontario, Canada) *(CASCON '14)*. IBM Corp., USA, 19–33.

[2] Ali Sajedi Badashian and Eleni Stroulia. 2016. Measuring User Influence in GitHub: The Million Follower Fallacy. In *Proceedings of the 3rd International Workshop on CrowdSourcing in Software Engineering* (Austin, Texas) *(CSI-SE '16)*. Association for Computing Machinery, New York, NY, USA, 15–21. https://doi.org/10.1145/2897659.2897663

[3] Natércia A. Batista, Michele A. Brandão, Gabriela B. Alves, Ana Paula Couto da Silva, and Mirella M. Moro. 2017. Collaboration Strength Metrics and Analyses on GitHub. In *Proceedings of the International Conference on Web Intelligence* (Leipzig, Germany) *(WI '17)*. Association for Computing Machinery, New York, NY, USA, 170–178. https://doi.org/10.1145/3106426.3106480

[4] Casey Casalnuovo, Bogdan Vasilescu, Premkumar Devanbu, and Vladimir Filkov. 2015. Developer Onboarding in GitHub: The Role of Prior Social Links and Language Experience. In *Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering* (Bergamo, Italy) *(ESEC/FSE 2015)*. Association for Computing Machinery, New York, NY, USA, 817–828. https://doi.org/10.1145/2786805.2786854

[5] Miguel del Fresno García, Alan J Daly, and Sagrario Segado Sanchez-Cabezudo. 2016. Identifying the new Influences in the Internet Era: Social Media and Social Network Analysis. *Revista Española de Investigaciones Sociológicas* 153 (2016).

[6] Linton C. Freeman. 1977. A Set of Measures of Centrality Based on Betweenness. *Sociometry* 40, 1 (1977), 35–41. http://www.jstor.org/stable/3033543

[7] Google. 2015. *Google Cloud Console BigQuery: GitHub Archive*. https://console.cloud.google.com/bigquery?project=githubarchive&page=project

[8] Georgios Gousios. 2013. The GHTorrent dataset and tool suite. In *Proceedings of the 10th Working Conference on Mining Software Repositories* (San Francisco, CA, USA) *(MSR '13)*. IEEE Press, Piscataway, NJ, USA, 233–236. http://dl.acm.org/citation.cfm?id=2487085.2487132

[9] Ilya Grigorik. 2015. *GH Archive*. https://www.gharchive.org/

[10] Derek L. Hansen, Dana Rotman, Elizabeth Bonsignore, Nataa Milic-Frayling, Eduarda Mendes Rodrigues, Marc Smith, and Ben Shneiderman. 2012. Do You Know the Way to SNA?: A Process Model for Analyzing and Visualizing Social Media Network Data. In *2012 International Conference on Social Informatics*. 304–313. https://doi.org/10.1109/SocialInformatics.2012.26

[11] Yan Hu, Shanshan Wang, Yizhi Ren, and Kim-Kwang Raymond Choo. 2018. User influence analysis for Github developer social networks. *Expert Systems with Applications* 108 (2018), 108–118. https://doi.org/10.1016/j.eswa.2018.05.002

[12] Yan Hu, Jun Zhang, Xiaomei Bai, Shuo Yu, and Zhuo Yang. 2016. Influence analysis of Github repositories. https://doi.org/10.1186/s40064-016-2897-7

[13] Eirini Kalliamvakou, Georgios Gousios, Kelly Blincoe, Leif Singer, Daniel M. German, and Daniela Damian. 2014. The Promises and Perils of Mining GitHub. In *Proceedings of the 11th Working Conference on Mining Software Repositories* (Hyderabad, India) *(MSR 2014)*. Association for Computing Machinery, New York, NY, USA, 92–101. https://doi.org/10.1145/2597073.2597074

[14] Jon M. Kleinberg. 1999. Authoritative Sources in a Hyperlinked Environment. *J. ACM* 46, 5 (sep 1999), 604–632. https://doi.org/10.1145/324133.324140

[15] Aman Kumar, Manish Khare, and Saurabh Tiwari. 2022. Sentiment Analysis of Developers' Comments on GitHub Repository: A Study. In *2022 14th International Conference on Advanced Computational Intelligence (ICACI)*. 91–98. https://doi.org/10.1109/ICACI55529.2022.9837754

[16] Antonio Lima, Luca Rossi, and Mirco Musolesi. 2014. Coding Together at Scale: GitHub as a Collaborative Social Network. https://doi.org/10.48550/ARXIV.1407.2535

[17] Huanhuan Liu, Xiaoqing Yu, and Jing Lu. 2013. Identifying TOP-N opinion leaders on local social network. In *IET International Conference on Smart and Sustainable City 2013 (ICSSC 2013)*. 325–328. https://doi.org/10.1049/cp.2013.1970

[18] Neo4j. 2022. *Neo4j Graph Data Platform | Graph Database Management System*. https://neo4j.com/

[19] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. *The PageRank Citation Ranking: Bringing Order to the Web*. Technical Report 1999-66. Stanford InfoLab. http://ilpubs.stanford.edu:8090/422/ Previous number = SIDL-WP-1999-0120.

[20] Ferdian Thung, Tegawendé F. Bissyandé, David Lo, and Lingxiao Jiang. 2013. Network Structure of Social Coding in GitHub. In *2013 17th European Conference on Software Maintenance and Reengineering*. 323–326. https://doi.org/10.1109/CSMR.2013.41

[21] Wikipedia contributors. 2020. Rich-club coefficient — Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=Rich-club_coefficient&oldid=983063532 [Online; accessed 10-October-2022].

[22] Wikipedia contributors. 2022. Borda count — Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=Borda_count&oldid=1113980432 [Online; accessed 10-October-2022].

[23] Wikipedia contributors. 2022. Erdős number — Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=Erd%C5%91s_number&oldid=1107735023 [Online; accessed 10-October-2022].

[24] Wikipedia contributors. 2022. H-index — Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=H-index&oldid=1111311697 [Online; accessed 10-October-2022].

[25] Wikipedia contributors. 2022. Spearman's rank correlation coefficient — Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=Spearman%27s_rank_correlation_coefficient&oldid=1112073400 [Online; accessed 10-October-2022].