Network Analysis of Strava Activity Patterns

Angelia Wang
Engineering Sciences and Applied Mathematics, Northwestern University

Introduction

Strava is a popular app used to track running (and other physical) activities. In this project, I analyze the network structure of my 500+ personal activities accumulated over the last three years.

Some of the questions this project sought to address:
How often are different segments covered? What is the distribution of these frequencies?
How unconnected are the runs? Do particular segments tend to identify isolated components, and is that because of geographic separation?

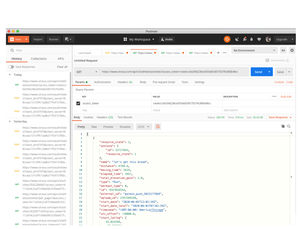
Methods and Data Structure

The Strava API allows limited (in time and amount) requests of any user's personal activities. I used the app Postman to request activity and segment data from the website, which returns information in JSON format.

Data collection pipeline is roughly summarized as follows:

1. Obtain and client secret (an alphanumeric code) and client ID from my personal strava account.
2. Use Postman to make a POST request using those two entries, which returns an access token (expires every six hours).
3. With the access token, request information on all of my activities. This returned a multi-hundred thousand line JSON file.
4. Write a Python script to interpret this file and make more data requests. For each activity, find the unique numeric ID. For each of those, make a separate request for the details of that activity, which includes a list of geographic segments.
5. Compile segment data and construct the network.

This network is bipartite by construction. Activity dates (in MMDDYYYY format) are one set of nodes, and segments (Unicode strings) are another.



b

Figure 1. (a) Postman interface (b) the massive JSON file

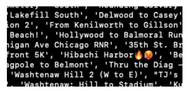


Figure 2. A segment containing emojis in the name wreaked havoc. Fixed by encoding names into Unicode.

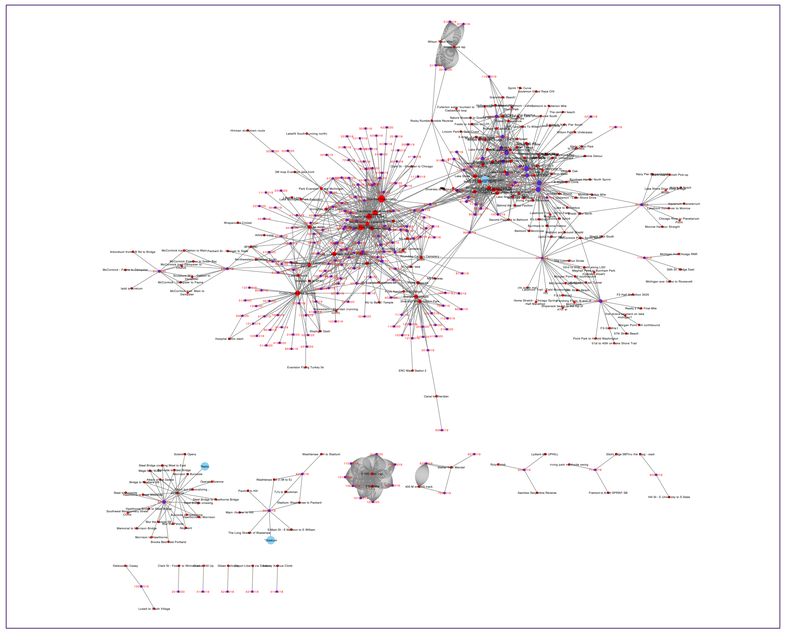


Figure 3. Full network, with segments in red and activity nodes in blue. Node sizes are correlated with the degree of the node. Largest connected component contains 370 nodes. Activity labels are the date (MMDDYYYY) they occurred and segment labels are taken off the Strava database. Made with Cytoscape.

Degree Distribution

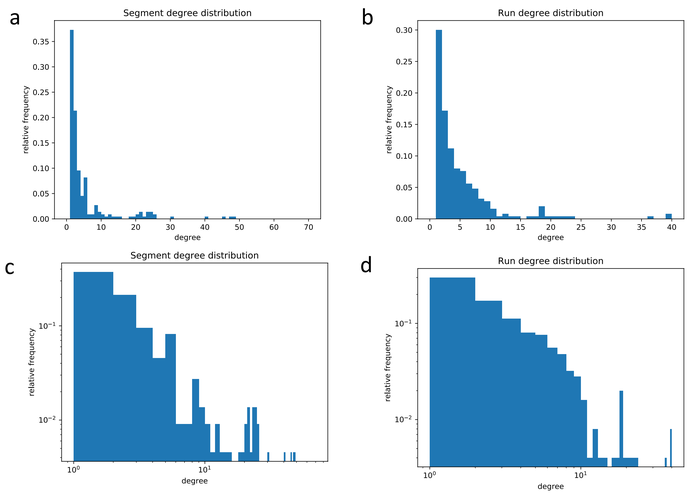


Figure 4. Degree distributions of segments (a,c) and runs (b,d) plotted on linear (a,b) and loglog (c,d) axes. Data resembles a power law distribution, but with the presence of high-degree nodes but finite datapoints, I cannot rule out the possibility it is scale free.

Further analyses

Bipartite network structure can further be analyzed by looking at its projection. For this graph, we have the following projection:

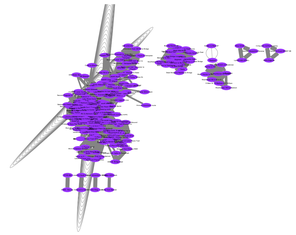


Figure 5. Projection onto segments. Edges weighted by Jaccard index, which measures overlap of nodes' neighborhoods.

The clustering coefficient of all nodes in the unprojected graph is 0.378, but the average C.C. in the projection is 0.781. A high average clustering coefficient suggests that most segments are connected to each other via at least one run.

Furthermore, the average path length is 4.335 (original) and 2.307 in the projection, suggesting that most runs/segments are connected through "hubs." In this context, hubs refer to popular segments or long runs that cover many segments.

Conclusions

There is a strong dependence of network structure on geographic location. Smaller connected components correspond to one-off runs done in a different city, races on unfamiliar routes, or paths with specific functions (e.g. public tracks). The large connected component encompasses nearly all Evanston and Chicago running segments.

Future directions include adding a time-dependent visualization tool that allows me to see how the network evolves chronologically.

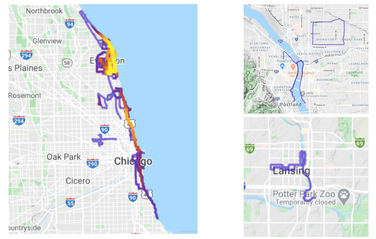


Figure 6. Heatmap shows strong preference for specific local routes and isolated runs in other cities.

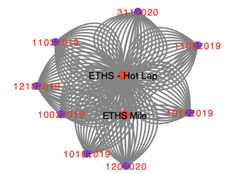


Figure 7. A smaller connected component Of a local high school track. It's disconnected because a track day explicitly will not include another run.

Contact

Angelia Wang
Engineering Sciences and Applied Mathematics
Northwestern University
angeliafwang@gmail.com
412.576.8324

Tools Used

This work is made by possible by: The Strava API, Postman, Python (networkx, matplotlib, csv), and Cytoscape.
Special thanks to Youtube user Franchyze923 for his informative videos on how to navigate the Strava API.
Special thanks to Jonathan O'Keefe, who wrote the code that generates geographical heatmaps.

All code I wrote can be found at https://github.com/angeliawang/strava_network