First Year Project #2
Maximizing Ad Revenue via Network Analysis

Michele Coscia

March 11, 2019

1 Goals

In this project, you will:

- Get basic familiarity with handling different graph data types;
- Learn pros/cons of different techniques to clean noise from network data to prepare it for the analysis;
- Reason about the relationship between data preparation task and business-supporting analysis.

2 Requirements

Same as the previous project's requirements.

3 Briefing

ITU Search has started its own new Ad Network. The customers are businesses who want to advertise on the search engine by having their links appear when a user queries the engine with specific combinations of keywords. The system has a limitation: once it sells a query to a customer, it cannot sell it to another one. So it cannot have two businesses sharing a query.

Management wants to increase revenue. One strategy they identified is query suggestion. Suppose you have a customer willing to pay $1 per impression for queries A, B, and C. Suppose that another customer who offers only $0.5 per impression, is interested in queries A, B, C, and D. Clearly query D must be similar to A, B, and C, since it interests them. Under the current bidding, ITU Search would only earn $3.5 per impression. Could we suggest query D to the first customer and bring the revenue to $4 per impression?

Your task is to tell ITU Search which queries to suggest to which client because similar clients requested them.

To complete your task you will have to face various issues. If the data were perfect, you could simply compare directly the queries requested by a customer with everybody else and find the most similar one. However, clients bid on queries they care little about, just to get something if somebody outbids them. Moreover, by having a fuzzier matching you could get more queries to suggest, while finding the most similar customer to a target customer might not generate enough suggestions.

A way to face these issues is to calculate the customer-customer similarity using network analysis. The customer-query structure is a bipartite network which you can project in a customer-customer similarity map using various techniques, and you can filter out the weakest connections to find groups of customers using network backboning and community discovery.

Here are some queries you should be able to answer:

1. Find the query that is used, in absolute count, the most in a community. Suggest it to the top 5 customers in that community, which have not used that query. Select them according to your own criteria.

2. Find any customer who bid only on two queries. Find at least three queries to suggest to them, among the ones which they didn't pick up.
3. Find the list of communities of non trivial size (with at least 60 queries) which is dominated by a single query (which is used by more than half the customers).

4 Hand-In & Oral

You must hand in the following.

1. The GitHub contributors log (in pdf).
2. The code scripts in plain text.
3. The project write-up.

Your 10 minute oral presentation should correspond to the structure of your write-up. However, you are encouraged to have slide headings that are more communicative. Your hand-in should be no longer than 3 pages (with 1.5cm margins and 11pt font size), and should consist of the following sections:

1. Introduction: to provide the context for the problem.
2. Methodology: to describe the methods – the combination of projection + backboning you chose and the motivation of why you think it was the most appropriate.
3. Results: to provide the technical results of your method over the data. I expect to find basic statistics of the final network generated (number of nodes, edges, etc).
4. Interpretation: the answers to the queries.
5. Concluding remarks: a couple of sentences summarizing the results of the project and indicating how the methods/data could be improved.
6. Disclosure statement (optional): to flag any important information regarding the group work that I need to know.