Analyzing Email Contact Graphs with Multiple Accounts

For this assignment, I decided to import all my email and analyze the corresponding contact graph. Due to Gmail archiving some of my previous email though, I was only able to grab all email from October 2008 (right after I started college) to the present (November 2013). This resulted in a graph with 1882 nodes and 3137 edges. Since the collection was done on only my email, there is an “assumption” made in the data that must be remembered. If two contacts have a relationship (directed edge) between each other, this is not indicative of all correspondence between those two contacts. Only the correspondence that I was included on is recorded in the data. To get a true sense of the email correspondence in the real world, it would require accessing the email of all other contacts in the graph.

![Initial graph view](image)

Figure 1 - Initial graph view

Headline 1: Identifying multiple email accounts

At first, I thought the network graph would be somewhat boring (Figure 1). It was obvious that there would be one node (me) in the graph that had a high degree due to it being my own email. Assuming there would be one cluster, I applied the Yifan Hu and Fruchterman-Reingold layout algorithm to identify and separate out any small clusters in the graph. I then applied a weight to each edge based on the quantity of emails between each contact in the graph. This produced an interesting result (Figure 2).
Figure 2 - Graph clusters

From this graph, it is easy to see there are ~5 “major” nodes that create clusters. Of the 5 clusters, there are 3 in particular that have many incoming edges with significantly high weight (circled above in Figure 2). This indicates that there is a large quantity of correspondence occurring with the email addresses. After zooming in and inspecting each one, I discovered that the email addresses were actually owned by me (my Gmail account, UMD CS account, and the account at my undergraduate institution). I did not originally account for the fact that my Gmail account is set up to handle multiple email accounts that signify "me". Fortunately, I have had this setup since I first began college, so was able to obtain a rich set of contact history for each email address dating back 5 years.

Headline 2: Distinguishing "real life" friendships from "professional-only" relationships
As a TA, it is important to keep the students you teach separate from your personal life (i.e. don't mix business with pleasure). Another thing I have learned is to never give out your personal email address (in my case, Gmail). Otherwise, students will inevitably bombard you with emails when project deadlines are approaching. Since Gmail handles all my email accounts, I normally do not pay attention to what email address the students use when contacting me, so it’s possible that my Gmail could have been inadvertently leaked to some students in the past. After realizing that I could observe multiple email accounts at once and see the community of users contacting me through a specific email, I colored all edges that connect students, whom I have taught in the past, red (Figure 3). I then colored all nodes/edges that correspond to TAs that I have worked with in the past, yellow (Figure 3). As a point of reference, I colored the node and all edges corresponding to the instructor (Larry Herman) that I have worked for in the past, light blue. My CS account ([jgbrad1@cs.umd.edu](mailto:jgbrad1@cs.umd.edu)) is colored green and my personal Gmail ([jgbradley1@gmail.com](mailto:jgbradley1@gmail.com)) is colored dark blue. I then applied the Yifan Hu and Fruchterman Reingold layout algorithms, which produced the graph in Figure 3. Edge width is calculated by the weight. As can be seen, all of my past students have contacted me only through my CS account, while the other TAs tend to contact me through both my Gmail and CS account. The graph also shows that Larry tends to contact me through my CS account more than my Gmail. These are the results I was hoping for. The TA’s and I tend to collaborate on google docs quite often, so it was to be expected for them to learn of my Gmail account. The other undergraduate students however seem to not know of my Gmail account.
Headline 3: Not having too many contacts who send out "mailing list" type of emails (which is a good thing)

Starting with the initial graph (Figure 1), I applied the Yifan Hu algorithm for a few seconds. Once clusters were distinguishable, I then applied the Fruchterman Reingold layout algorithm to spread out the clusters. I then colored the nodes by their degree on a green-to-red color scale. The higher in-degree nodes have more green, while the higher out-degree nodes are colored red. In the context of email, this translates to saying the more incoming email a contact receives (high in-degree), the greener they are. If a contact sends out more email than they receive (high out-degree), they are colored red. Edge thickness is denoted by the weight

\[
weight = \frac{\text{# of emails exchanged between source node and target node}}{\text{total # of emails sent out by source node}}
\]

I took one final step and encoded the node size to reflect the out-degree (i.e. low out-degree will translate to small nodes, while a high out-degree will make the node larger). Looking at Figure 3, it is easy to see there are essentially 3 “interesting” nodes. The two red nodes at the top of the graph belong to the department chair and department secretary of my undergraduate institution. The other red node at the bottom of the graph belongs to Larry Herman, an instructor in the CS department at UMD, who I have TA’ed for, for the past 3 semesters. The graph shows that these 3 contacts sent out more email than they received (with me included in the emails) due to the color/size of their corresponding node and by inspecting the edge
thickness of their outgoing connections, it is easy to see that most of the edges have around the same weight. This means that when the contact sent out an email, they normally sent it to a whole group of people vs. one person. This type of behavior is normally found in contacts that act as mailing lists, sending out general announcements. From my personal experience, I can confirm this type of behavior for the department chair and secretary at my undergraduate institution. Although one-on-one personal emails were exchanged with both people over the years, most of the email I received from them was of the mailing-list type. This behavior can also be observed with Larry Herman due to the fact that he tends to send out one email to all the TA’s helping him, in order to save time and make announcements that apply to everyone.

Critique:
I used Gephi for this project. It had many useful features, including the ability to import email. Although it lacks in the number of import tools when compared to NodeXL, I believe it does a better job at displaying large graphs and processing them than NodeXL. I use a Mac so running Windows in a virtual machine and using NodeXL resulted in poor performance and slow interactions with the data. Gephi on the other hand is open source, and can be installed directly on Mac. I did run into a few problems initially during installation. These seem to stem from the fact that Apple recently released a OS update (Mavericks) and developers have not had time to apply patches to Gephi’s source code. I thought that the “Data Laboratory” in Gephi made it very easy to modify the data when necessary. It was also very easy to group nodes together and aggregate their edges. With NodeXL, it seems that after you group data together, you lose the original dataset and cannot return to it, whereas Gephi allows you to ungroup nodes whenever you wish. Gephi seemed to provide more filtering capabilities than NodeXL and it was easier to locate the various filters in the UI than NodeXL.