Email Response Time Analysis

An analysis of email response time between a user and his/her contacts.

Meng MENG
mm8134@nyu.edu mm8134
Runbo GAO
rg3099@nyu.edu rg3099
Yanlin ZHUANG
yz2833@myu.edu yz2833

Project page (on Github):
https://github.com/TonnyKen/IVFinalProject/blob/master/README.md
Video: https://vimeo.com/19688438
Working demo: https://tonnyken.github.io

What is the problem you want to solve and who has this problem?

To find out the regular pattern of email response behavior of an email user and his/her contacts.

Those who hold an email address and what to know his/her relationship with all the contacts might have this problem, as well as their habits to respond an email.

What are the driving analytical questions you want to be able to answer with your visualization?

What is the relationship between response time and the specific contact (One may often respond a close friend or the boss faster, but respond a stranger slower). That might indicate who are closer contacts and who are not.

What is the relationship between average receiving time from and replying time to a specific contact? And how the response time looks like when comparing with each other?

For a specific contact, what is the email quantity distribution over response time? (For a specific contact A, one could reply 3 emails in 5 minutes, 2 emails between 5 and 10 minutes, 4 emails between 10 and 15 minutes…)

What does your data look like? Where does it come from? What real-world phenomena does it capture?

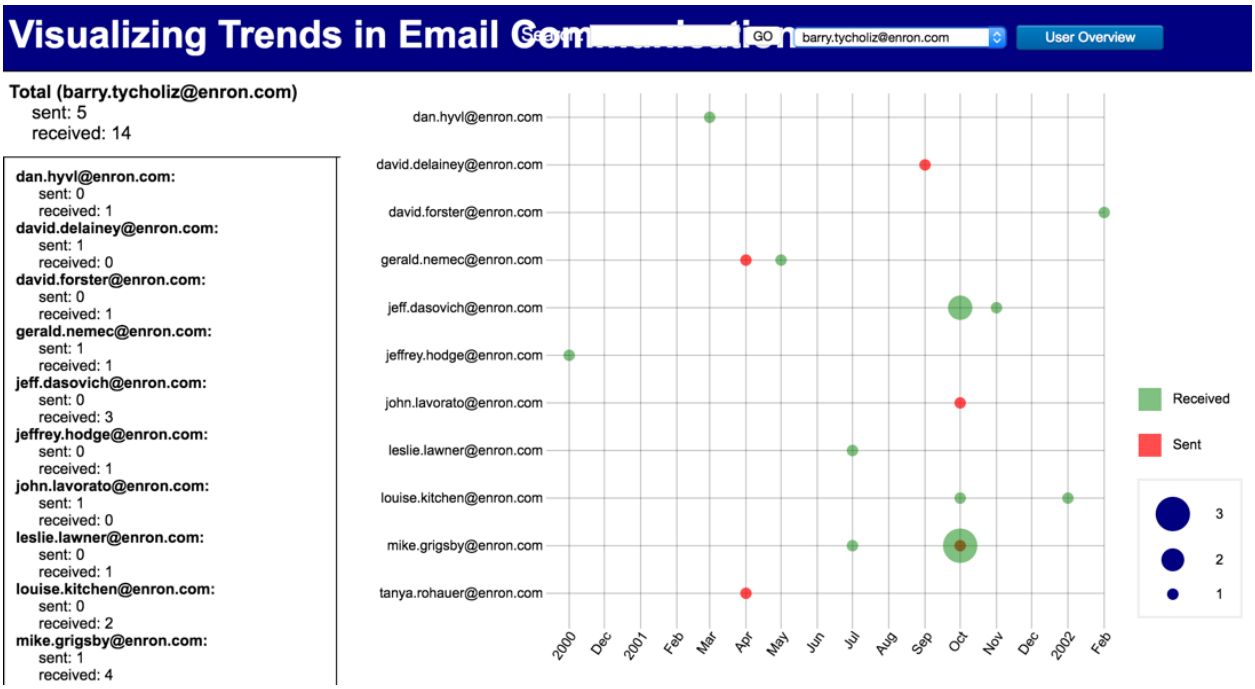
Attribute Name	Attribute Type	Meaning	Values	Derived
To_address	Categorical	Contact's address	N/A	NO
FromCount	Quantitative	How many emails received from this address	Non-positive Integer	From date
ToCount	Quantitative	How many emails replied to this address	Non-positive Integer	From date
FromAverage	Quantitative	Average receiving time	Non-positive Float	From date
ToAverage	Quantitative	Average replying time	Non-positive Float	From date
TotalCount	Quantitative	FromCount + ToCount	Non-positive Integer	From date

Attribute Name	Attribute Type	Meaning	Values	Derived
To_address	Categorical	Contact's address	N/A	NO
From<120	Quantitative	Quantity of email with ReceiveTime<120min	Non-positive Integer	From date
From120~240	Quantitative	Quantity of email with ReceiveTime120~240	Non-positive Integer	From date
From240~360	Quantitative	Quantity of email with ReceiveTime240~360	Non-positive Integer	From date
From360~480	Quantitative	Quantity of email with ReceiveTime360~480	Non-positive Integer	From date
From480~600	Quantitative	Quantity of email with ReceiveTime480~600	Non-positive Integer	From date
From>600	Quantitative	Quantity of email with	Non-	From

		ReceiveTime>600	positive Integer	date
To<120	Quantitative	Quantity of email with ReplyTime<120min	Non-positive Integer	From date
To120~240	Quantitative	Quantity of email with ReplyTime120~240	Non-positive Integer	From date
To240~360	Quantitative	Quantity of email with ReplyTime240~360	Non-positive Integer	From date
To360~480	Quantitative	Quantity of email with ReplyTime360~480	Non-positive Integer	From date
To480~600	Quantitative	Quantity of email with ReplyTime480~600	Non-positive Integer	From date
To>600	Quantitative	Quantity of email with ReplyTime>600	Non-positive Integer	From date
FromAverage	Quantitative	Average Receive Time	Non-positive	From date
ToAverage	Quantitative	Average Reply Time	Non-positive	From date

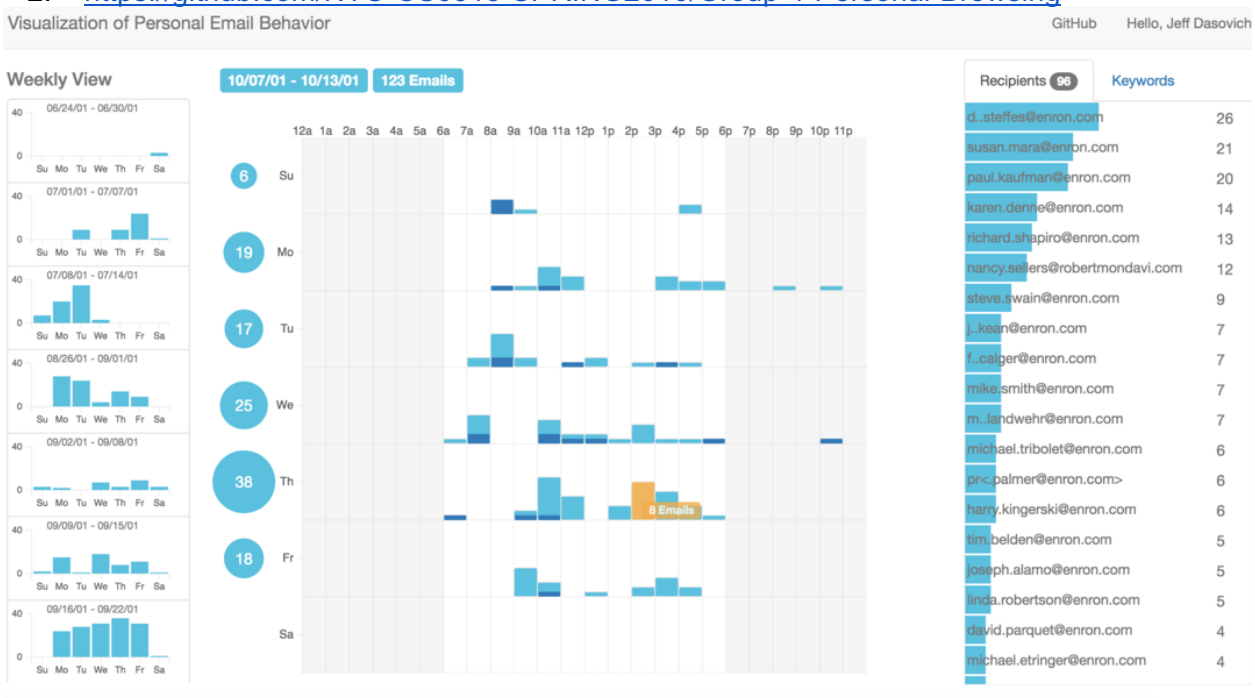
What have others done to solve this or related problems?

1. http://nyu-cs6313-fall2015.github.io/Group-4/



This project shows how many emails were received and sent by each user on different quarters of a year. Both this project and our project need to calculate the number of received and sent emails. But it didn't focus on the responding time.

2. https://github.com/NYU-CS6313-SPRING2016/Group-4-Personal-Browsing



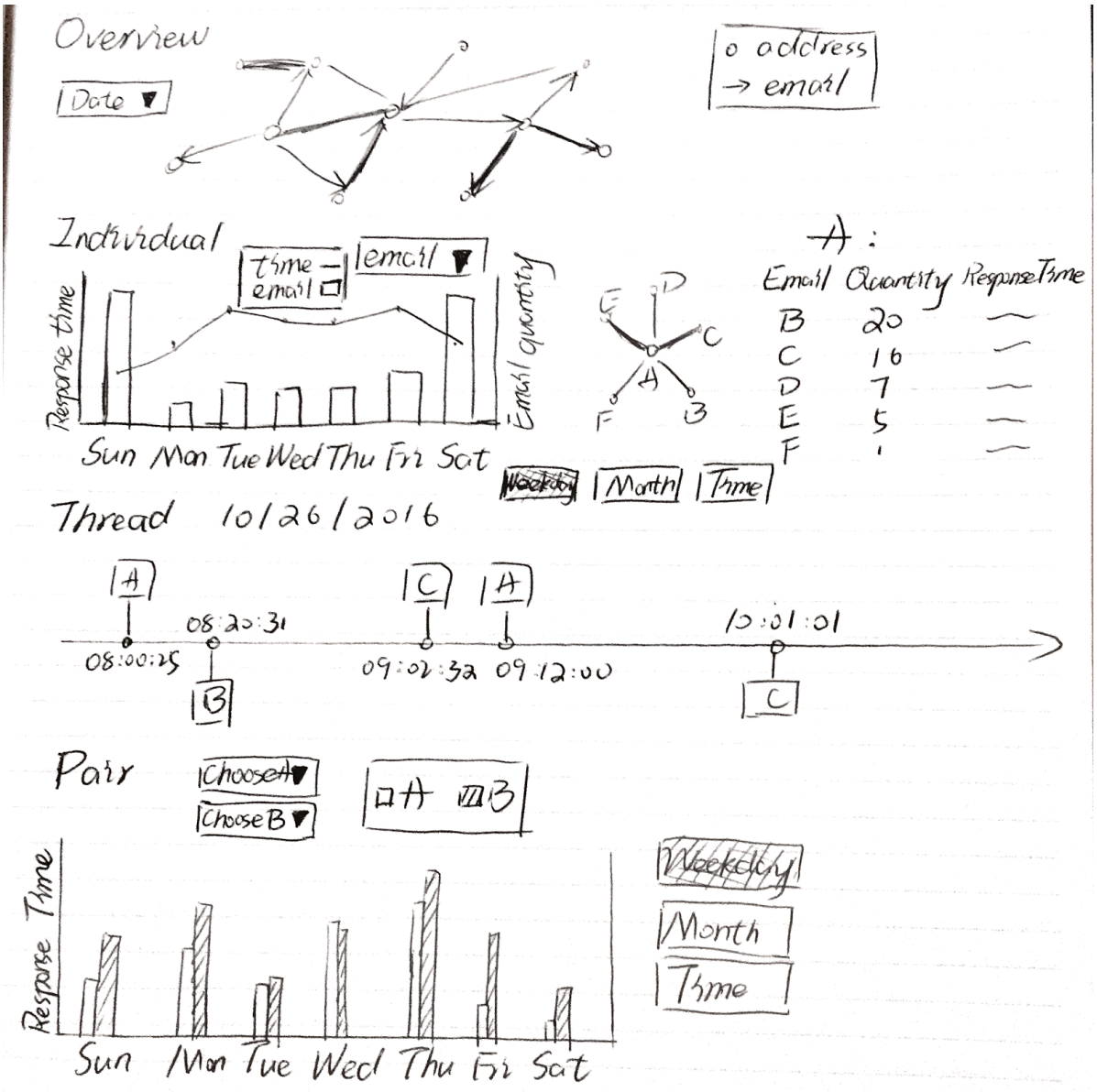
Search for related works and explain how they are related to your work. Add links and images of

projects that have solved similar or related project. For each one add a sentence explaining what the project is about and how it related to your project.

Design Iterations

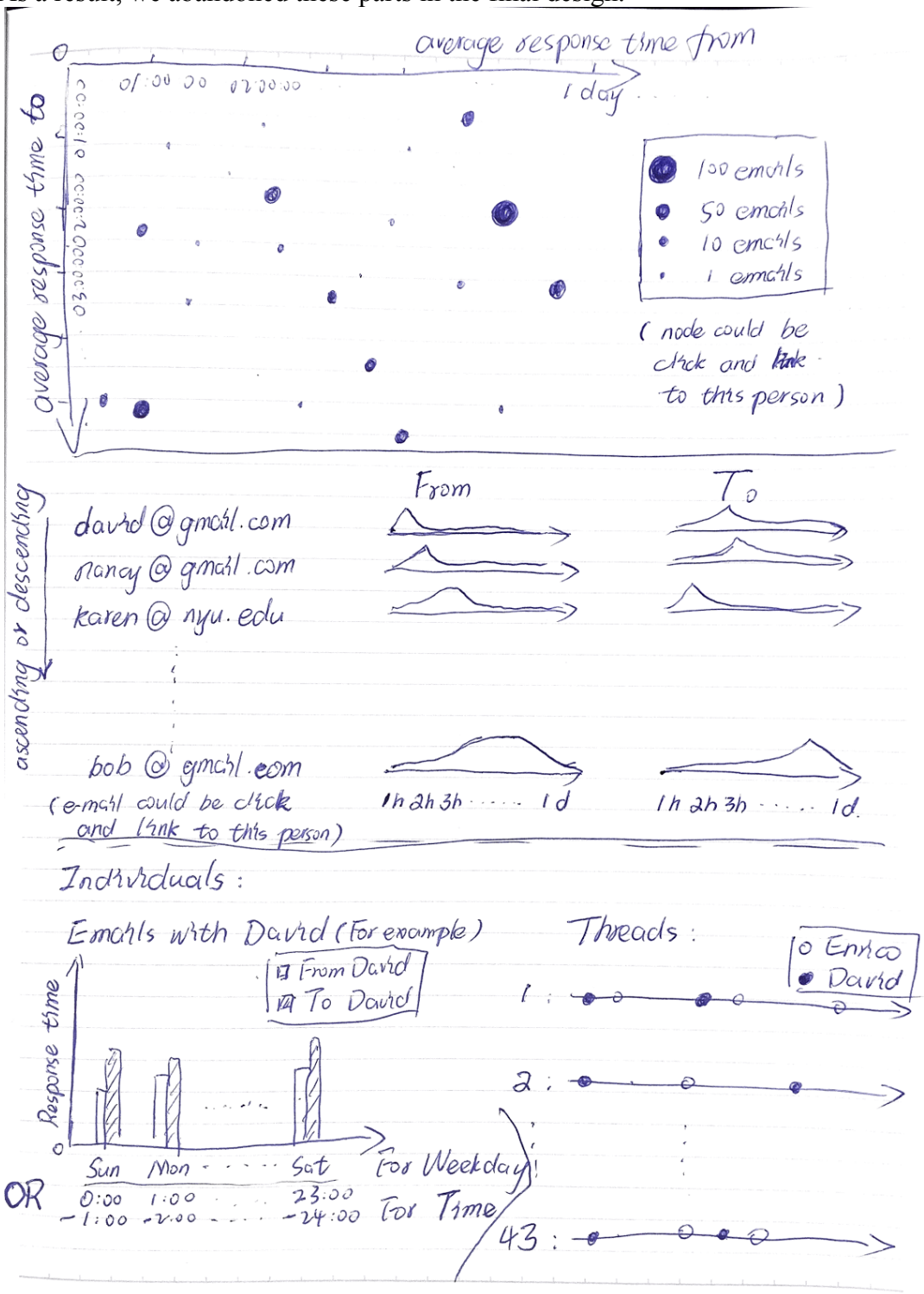
First Mockup:
In the overview part, the network graph shows the relationships between email addresses in a given date(users can choose by using drop down icon). The size of the node represents how many emails this address has sent or received (users can click the button to choose). The size of the edge shows the average responding time of an email from one address to another. Clicking on the node links to the detailed information of an individual address.
In the individual part, there are two subparts. In the first one, user can choose an individual(by using drop down icon) to show in the chart the average responding time and total emails on a weekday, in a month, or in an hour. In the second one, users can know how other address interact with the chosen individual.
In the thread part, users can choose a thread(by using drop down icon) to show the timeline, which indicates the sending time of an email.
In the pair part, the bar chart shows two individuals'(users can choose by using drop down icon) responding time to each other. Users can choose to show the average responding time on a weekday, in a month, or in an hour.

However, the dataset is about the emails from one email box, which means all the emails are related to a specific person. So the network is a bad design. What's more, the Pair part is not a very effective for the same reason. As a result, we abandoned them in the next mockup.



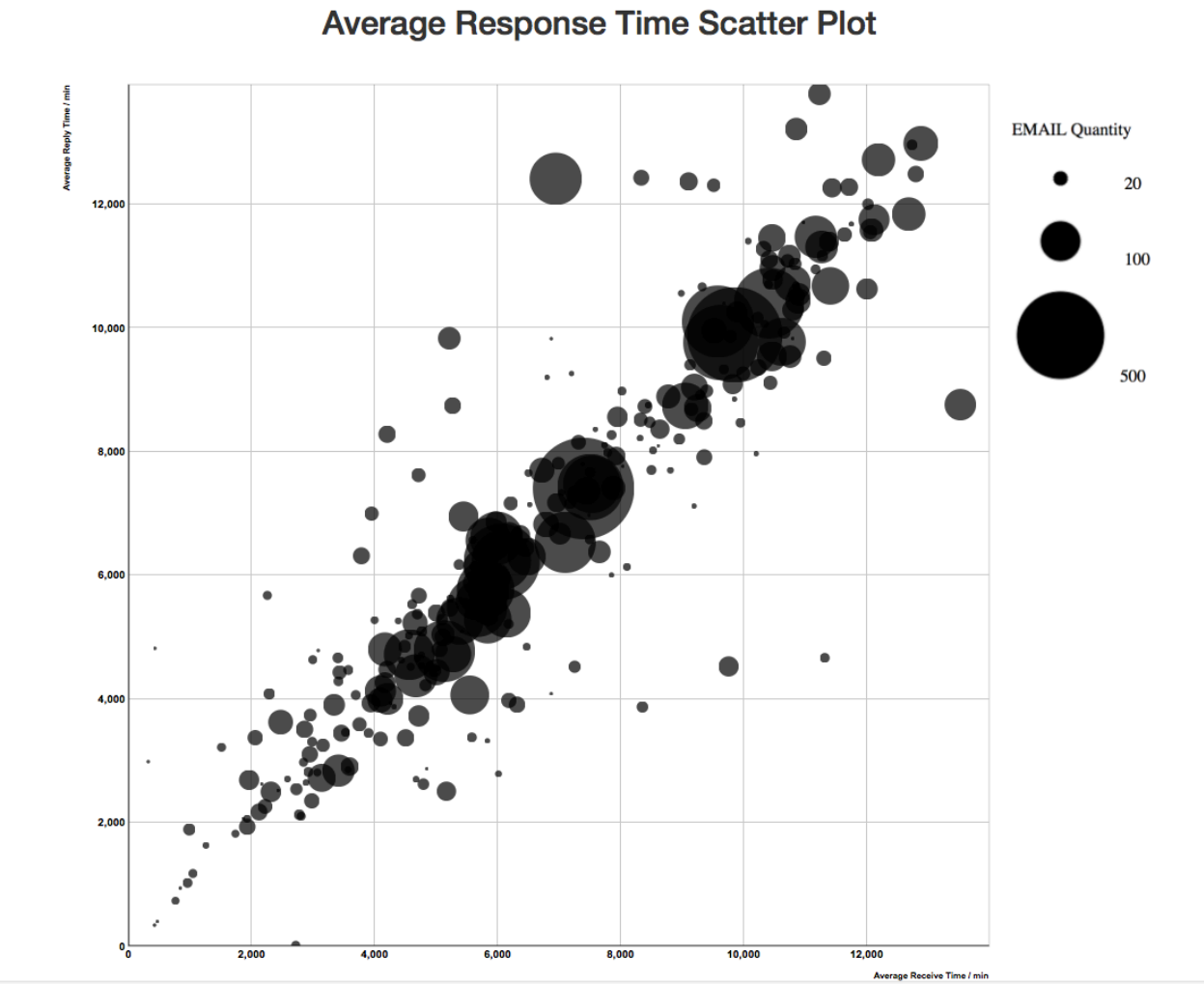
Second mockup:
At the top and middle of the mockup, overall information are visualized. The scatter plot shows the relationship between response time from an individual and to an individual. The list of line chart shows the response time distribution sorted in ascending or descending order, and by FROM or TO.
At the bottom, individual information is showed after clicking a person whether in the scatter plot or in the list. Grouped bar chart shows response time in different time and time axis show threads with this individual involved in.
However, most people tend to respond email slowly during weekends and night, so the analysis of how response time relates to weekdays and time is ordinary. What's more, some people respond to an email after a very long time like a couple of months, which causes the difficulty of

drawing time axis about him (It would be hard to distinguish points because of a large outlier.) As a result, we abandoned these parts in the final design.

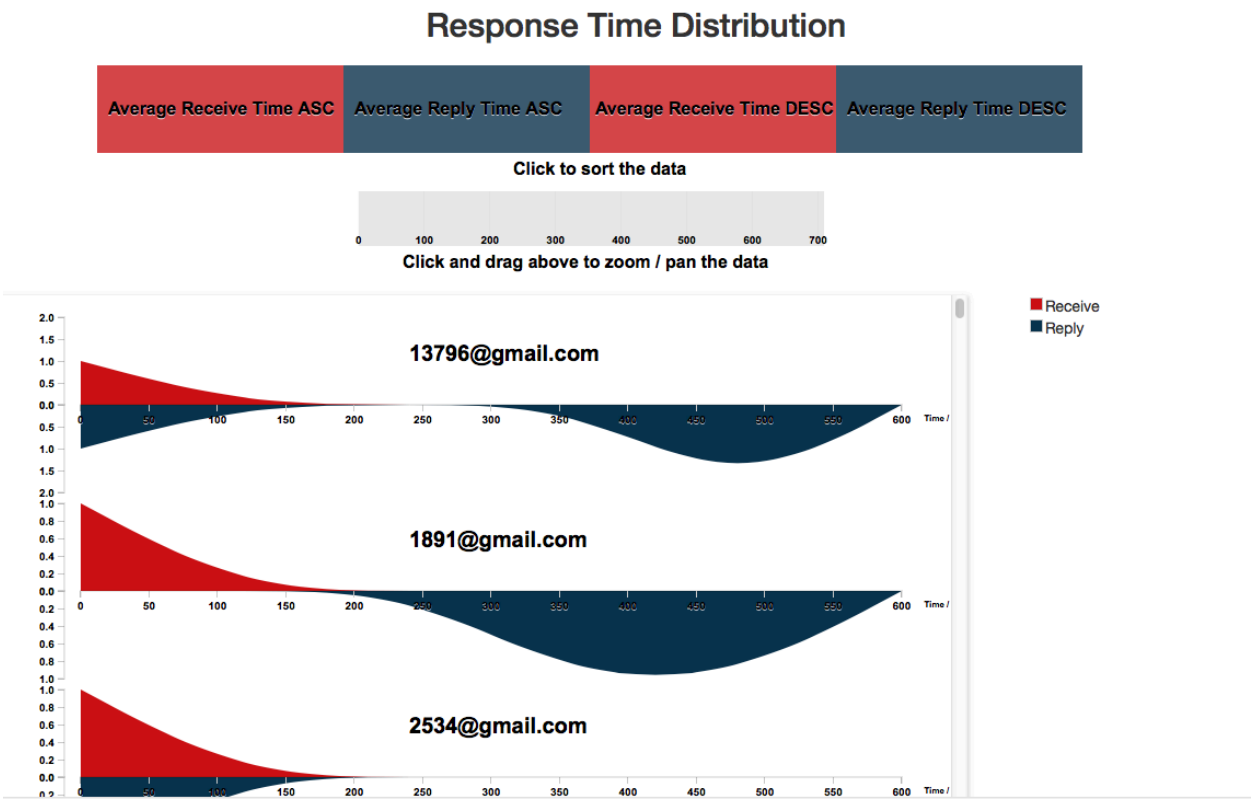


Final Visualization

In the first part, a scatter plot of all contacts is showed, with x-axis for average receive time, y-axis for average reply time and the size of bubble for emails involving with this contact.
User can: (1) zoom in and out to see the details of the part they are interested in; (2) put the cursor on bubble to show the detailed information of a contact; (3) click the bubble to show the violin graph in the following contained of the second part.

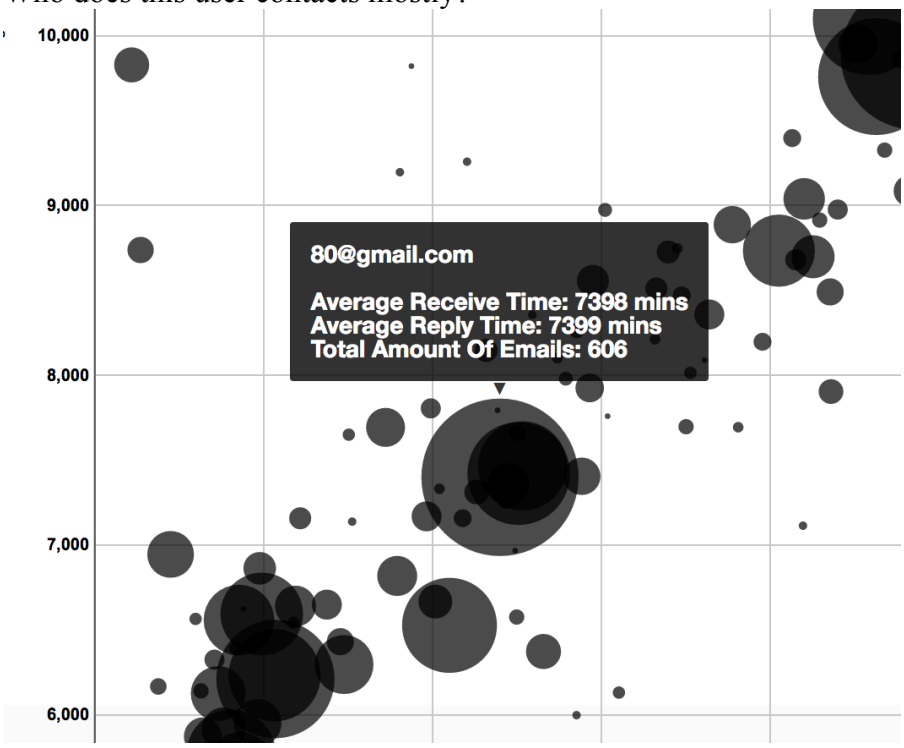


In the second part, a violin plot list of the response time distribution is showed.
User can: (1) click the button to sort the list by reply time or receive time in ascending order or descending order; (2) click and drag the bar to choose to show the subpart of the data in an order mentioned above (For example, show the top 50 contacts of the data in ascending reply time order); (3) click the bubble in the first part to show the violin plot of that contact.



Findings

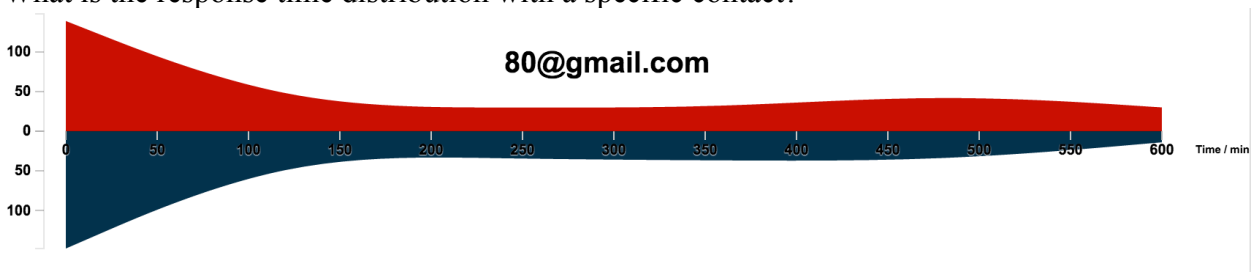
(1) Who does this user contacts mostly?



80@gmail.com is the most frequently contact.

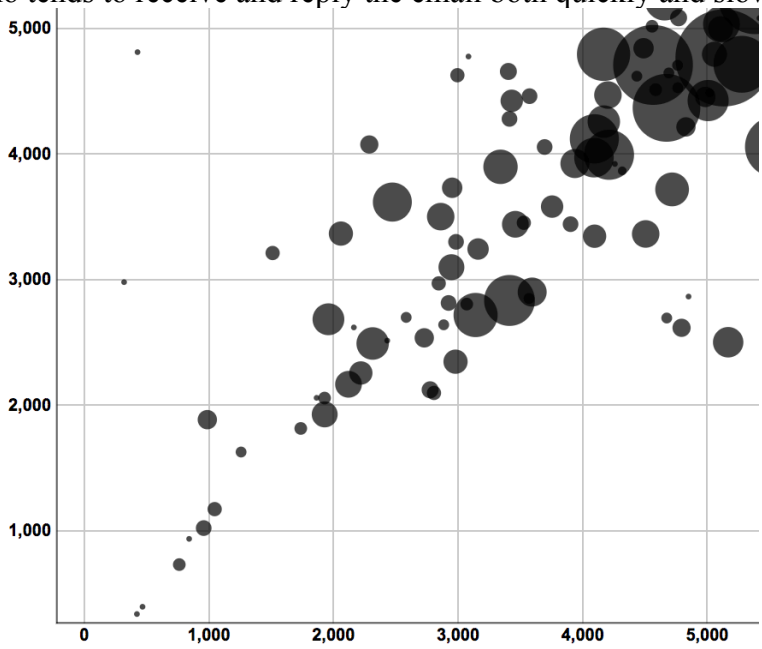
Now that I know the average response time between the user and this address, what's the response time distribution?

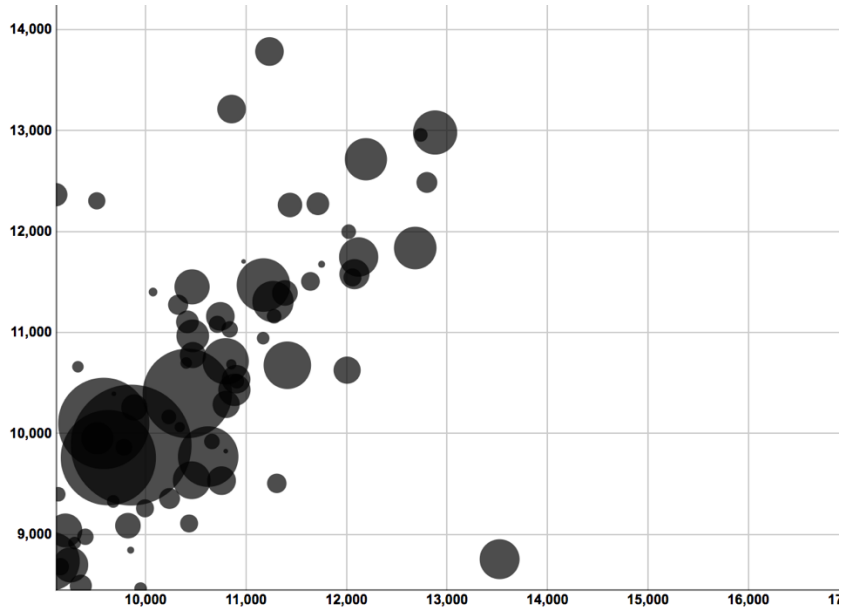
(2) What is the response time distribution with a specific contact?



It seems that most of the emails involved with 80@gmail.com are with the response time less than 50 minutes.

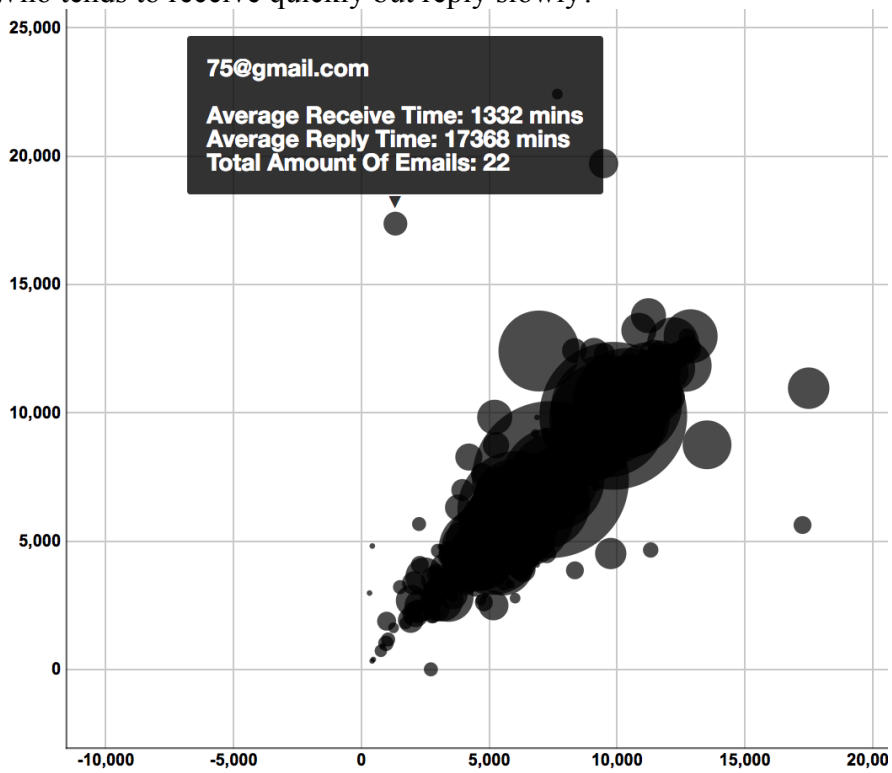
(3) Who tends to receive and reply the email both quickly and slowly on average?





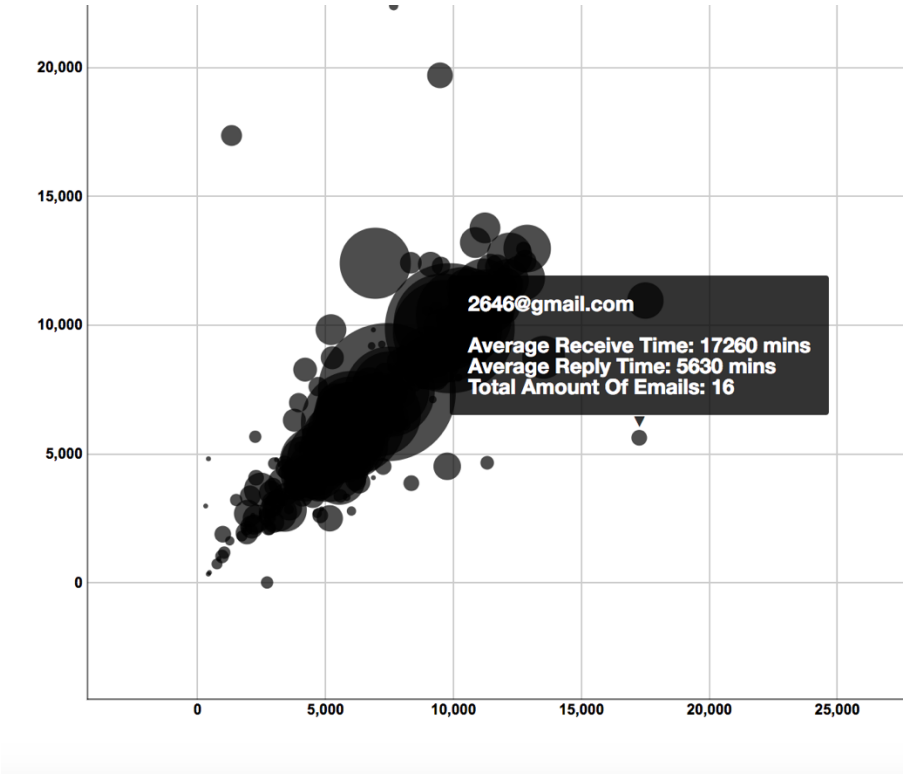
Bubbles on the bottom-left and top-right corner shows them respectively.
They might be the contacts who the user is getting along well with and not very familiar with respectively.

(4) Who tends to receive quickly but reply slowly?



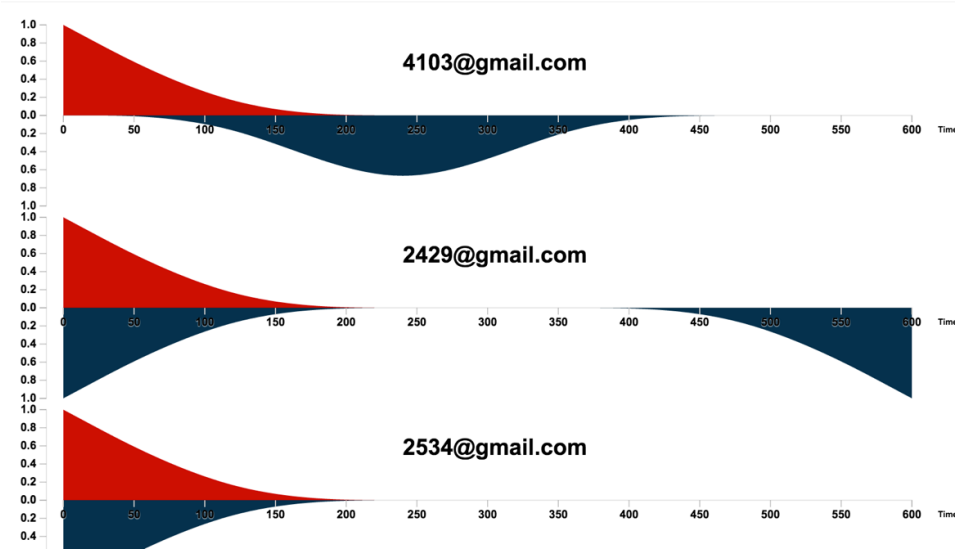
As the image shows, 75@gmail.com is a good example. The user receives from this address very quickly, but tends to reply slowly.

(5) Who tends to reply quickly but receive slowly?

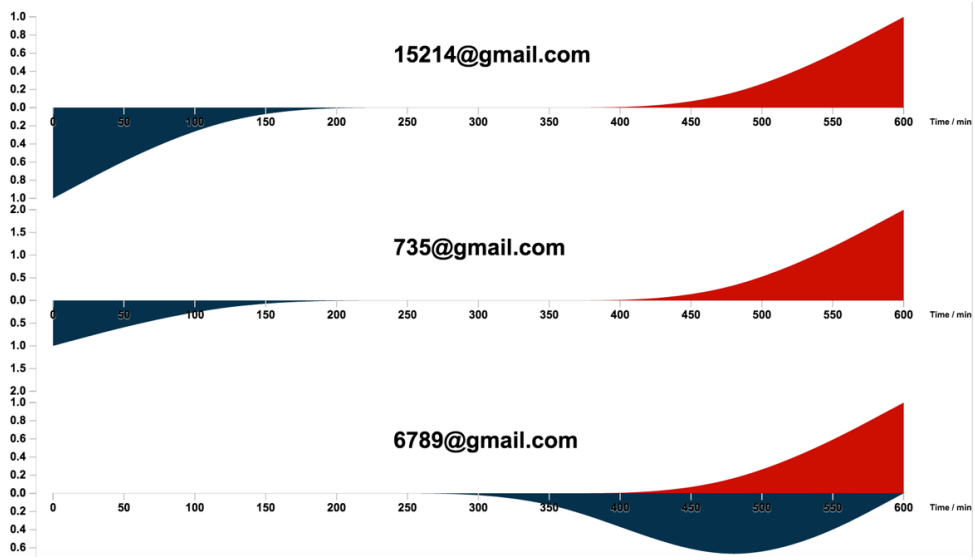


As the image shows, 2646@gmail.com is a good example. The user receives from this address very slowly, but tends to reply quickly. Both (5) and (6) are interesting because they show unequal relationship between the user and a contact. Maybe only the user knows the reason.

(6) Who does this user receive email quickest or slowest from? And what are the response time distribution of them?

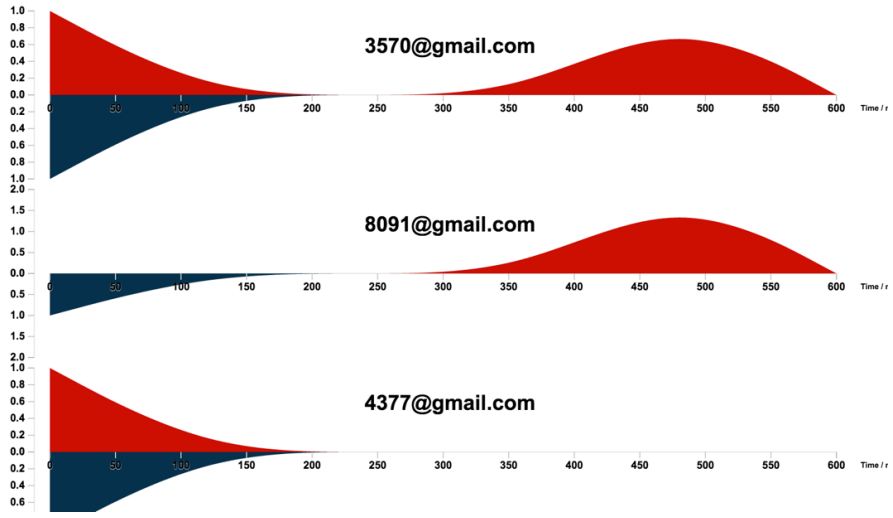


4103@gmail.com 2429@gmail.com 2534@gmail.com are the top3 quickest address to receive from.(use the sorting button)

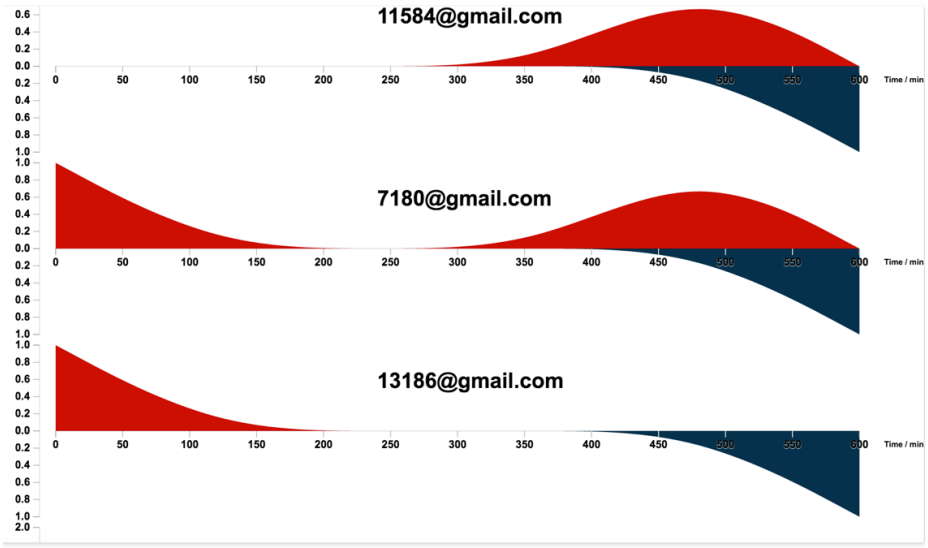


15214@gmail.com 735@gmai.com 6789@gmial.com are the top3 slowest address to receive from.(use the sorting button)

(7) Who does this user reply email quickest or slowest to? And what are the response time distribution of them?



3570@gmial.com 8091@gmail.com 4377@gmai.com are the top3 quickest address to reply to.(use the sorting button)



11584@gmial.com 7180@gmail.com 13186@gmai.com are the top3 slowest address to reply to. (use the sorting button)

Limitations and Future Works

Limitations:

(1) The legend about size of scatter plot is hard to draw. We didn't come up with a good way so we made an image and paste it (but we drew it precisely according to the radius and email amount).

(2) We can show the violin chart of a contact by clicking the bubble but can't do it the opposite way.

(3) The elements of the visualization are a little large so the user need to scroll to go to the scatter plot or the violin-chart container.

Future works:

(1) Learn to make the legend about size.

(2) Add the action after clicking the violin chart, which shows the corresponding bubble.

(3) Adjust the size of each elements to fill in just one page