Peter Dewire  
December 16, 2015  
CPSC 490  
Advisor: Prof. Holly Rushmeier

iMessage Visualization

I. Introduction and Data

Electronic communication is a phenomenon that has quickly and dramatically transformed the way people interact. Only a few decades ago, almost all communication occurred either in person or by hand-written letters. Now rarely a day goes by that the average American does not use some form of electronic communication, be it an e-mail, text or tweet. Based on the few different data sets on which I tested my project, a college student might send and receive between 110 and 170 texts per day.

In my project, I used Javascript’s d3 library to visualize the massive amount of data in the hundreds of thousands of texts that are stored by the iPhone’s and Mac’s Messages app. The data for Mac Messages are stored in ～/Library/Messages/chat.db and include fields such as message date and time, whether the message was sent or received, the text of the message, the length of the message (in characters), and the contact who sent the message, among other things. The same data are stored for iPhone backups in ～/Library/Application Support/MobileSync/Backup/.

The next section will present the seven visualizations that I created. Section III will provide some thoughts on the nature of text message data and the usefulness of each visualization, as well as comparisons between visualizations of different people who tested the code and feedback from those testers. Finally, section IV will conclude and give credit to those who helped with the project and the original authors from whom some of the visualizations are adapted.

II. Visualizations

The seven visualizations can be broken up into three different groups. The scatter plot and stream graph were the first two visualizations created, and both are simple, broad in scope, and intuitive. The word cloud, word tree, and word bubbles are all designed to give insights into the uses of individual words, with the word cloud being the simplest and least informative and the word bubbles being the most complicated and most informative. Finally,
the cross filter and parallel coordinates present aggregate information that can be filtered down to give more specific insight.

**Scatter plot**

The first visualization gives a quick and simple overview of who you text the most and who texts back the most. Each blue dot represents one of your contacts; the x-axis measures the messages that you send, and the y-axis measures how much you receive. The unit of measurement for both the x- and y-axes (and, generally, throughout project) is individual characters (including spaces). Thus, if a contact were on the line $y = x$, that would mean that for each character of text that you send to that contact, you receive one character in return. By hovering over a dot, you can see the individual statistics of that contact.

The scatter plot also provides a table with summary statistics for the 25 people you communicate with the most via iMessage. There is also a search feature, so that you can find contacts without hovering over each dot, and discover who is in the dense cluster that most users see in the bottom left corner (people with whom you exchanged only a few short messages, of which there are often many).
Stream Graph

Another graph that gives a basic overview of your texting habits is the stream graph, which shows how the frequency of your communication with your top contacts varies over time. On the x-axis is the date, and each different shade of blue in the visualization represents a different contact. The height of the shade on a certain date tells you how many characters you exchanged with that contact on that day. Thus, this graph allows you to see how your Messages usage varies over time both in aggregate and by contact.

Word Cloud

Another simple visualization, the word cloud shows which words you and your friends use the most in your text conversations. The size of the word in the cloud is proportional to how often the word is used in both your sent and received texts. Many words, such as “a,” “the,” “that,” “I,” “is,” are taken out of this visualization as they are to be expected and add little value or information to the cloud. Many common and obvious words, such as “yeah” (the largest, and thus most frequently used, word in the cloud below) do still appear, as it is difficult to account for all such words.
There is also some potentially interesting information that can be gleaned from such words; for example, people differ in their standard affirmative – as seen here, mine is “yeah,” but others may use “yes,” “yea,” or “ya.” These nuances will be discussed in more detail in section III.

**Word Tree**

The word tree, like the cloud, shows which words you and your friends use the most, but it also shows how your words commonly fit together. On the right side of the tree are data from messages that you have sent, and these are ordered from beginning to end.
(meaning that parent nodes are words or characters that come before their children). On the left side are data from received messages, and these are ordered in reverse (meaning the parent nodes are words or characters that come after their children). The extra information available in the tree comes at the cost of simplicity in its visualization: the tree is massive and the insights that it has to offer are not as immediately available as are those of the word cloud. Shown on the right is a zoomed out picture of the entire tree. Of course, no information can be gathered by looking at the entire tree, so it is necessary to examine it in parts. Shown on the left is a small section of the tree (the top of the rightmost section of the full tree), which shows the word “yeah” (this word is on top because it is again the most used word). If a node shares an edge with “yeah” (such as “I” in the example above), that means that you have sent messages that contain the phrase “yeah I...”. Performing a walk down the tree produces a full message, meaning that nodes that have many children are sentence constructions that often begin your messages.

**Word Bubbles**

The third and final word-centric visualization is word bubbles, which shows the proportions in which you and your contacts use your most commonly exchanged words. By
default, the visualization shows you the 30 most used words (which are also the 30 largest words in the cloud). Each bubble is split into blue, which represents the percentage that you use the word, and red, the proportion your contacts use the word. Excerpts below the visualization (which extend much further than in the example here) show specific examples of messages that contain the word. The search bar above the bubbles allows you to add lesser used words and phrases (e.g. “computer science”) to the visualization, a feature that can be especially useful when searching for a specific conversation.

**Cross Filter**

One of the most informative visualizations, the cross filter gives insight about who you communicate with and the day of week, time of day, and date in history that you do so.
October, 2012 (in this example, the messages suggest that she was driving to Yale for a visit). As shown in the example, the different fields adjust as you apply filters; thus, when you select a certain day of the week or range of dates, the contact section will adjust to show you who you were communicating with most at that time. Thus, the cross filter provides insight into your texting habits at varying scopes: with no filters selected, you can see generally who you communicate with most and when. As more filters are applied, you can see more granular detail, down to specific events or conversations.

Parallel Coordinates

Similarly to cross filter, parallel coordinates allows you to apply filters in order to
narrow down your search. The main difference between the two visualizations is their focus: parallel coordinates is concerned more with contact data, whereas cross filter is more concerned with individual message data. The visualizations also differ substantially in how they look and the way in which they are used.

Each axis in the parallel coordinates can be “brushed” to show only a subset of the data. So, as an in the example above, you could look only for contacts with whom you have communicated in the last 200 days, have exchanged over 50,000 characters, and have exchanged between approximately 1,000 and 5,000 messages.

III. Analysis, Comparisons, and Feedback

One of the main lessons of this project relates to the unstructured nature of the actual text of these messages. This is especially apparent when looking at a visualization like the word tree. Although the most common words, such as “yeah” and “thanks,” exhibit some structural standards and create nicely balanced branches, the vast majority of words appear only once or twice at the beginning of text messages. Therefore, the majority of the tree looks like this image, taken from the very bottom of the “sent” side, where each node has only a single child. Because of the virtually infinite variations and combinations by which
words can form language, many visualizations, such as the word tree, are not well suited to perform this sort of analysis. The most successful visualizations in this project were the ones that were based on information other than the texts itself, such as the cross filter, which took advantage of differences in time and contact. Also, it is easier and often more informative to look at differences between the characteristics of different contacts than individual messages because messages vary so widely, and because the vast majority of them are simple one or two word phrases such as “yeah” or “almost there.” When examining characteristics of a contact, the many messages exchanged with that contact are aggregated so it is easier to see average differences. For this reason, I changed parallel coordinates from originally having individual messages as the base unit (as shown here) to having contacts as the base unit and looking at more aggregate data, such as number of texts and average response time. With further study and more in-depth analysis, however, there may be very interesting insights that available from visualizations based on individual messages.

It is especially interesting to compare the results of different people’s data sets. For example, the following scatter plot is based on the data of a fellow student who is in a long-term relationship. The plot shows that he communicates with his girlfriend, the dot circled in red in the top right corner, far more than any of his other contacts. Although this is to be expected, it is interesting to see this explicitly in the data. Also, the sheer proportion of texts
sent to his girlfriend may be surprising (over 7.5x more than his next highest contact). Looking at the word cloud from the same student, we see that one of the most commonly used words is “babe,” which differs from those users who do not have a significant other. While these observations are somewhat obvious, users of the app enjoy seeing them spelled out in the visualizations. In fact, the main feedback from the testers is that they would like to use the app primarily for fun as opposed to for serious analysis or questioning. They enjoyed, for example, picking a certain event (such as high school graduation) and going back to that time period and seeing what they were talking about with their friends. Thus, the main improvements they suggested were to user experience: e.g. faster load times, a more intuitive UI, and, of course, more visualizations. Ideally in the future, the project could be run as a simple web application to increase the ease with which it could be shared.

IV. Conclusion and Credits

Overall, the project went quite smoothly – all of the definite deliverables that I outlined in the initial proposal were created. Unfortunately, there was little time left for the advanced deliverables (which were also overly ambitious) such as sentiment analysis and intelligent categorization of conversations. Many of these “stretch goals” would serve as great opportunities for future work, and I hope to continue working on the project myself. The feedback from those who tested the project was very positive, and everybody greatly enjoyed learning about their texting habits and playing with the visualizations. I also hope to distribute the project to a wider group of people and have people throughout the school try it out.

The major challenges that I encountered were collecting the data, acclimating myself to Javascript, and choosing the proper visualizations as well as what to express in each. The data for the messages was difficult to find at first, and it was in a complicated format (for instance, the contacts were stored in a different database than the text messages). Also the data had to be molded into different formats for each visualization, which was often challenging due simply to the size of the data that was being used (over 160,000 messages
over the course of five years). d3 was the most appropriate tool for the project, as Prof. Rushmeier pointed out early on, but I had never used Javascript and had little experience with web programming, so there were several challenges to overcome both in learning the syntax of a new language as well as getting comfortable in a web development environment. The final major challenge, which is a great skill of data analysts, was determining which visualizations would best serve my purposes. Most of the visualizations underwent several iterations before taking on their final form – seeing these iterations and making decisions about the best way to represent certain phenomena was one of the most valuable and didactic parts of the project.

I would first and foremost like to thank my faculty advisor, Professor Rushmeier, for the guidance, suggestions, and support throughout the semester. I would also like to thank student testers Kai Takahashi, Georgia Blatchford, Thomas Dembinski, Liam McClintock, Charlotte Belling, and Amin Mirzadegan, who all helped find bugs in the projected and provided great feedback. Finally, I would like to give credit to Mike Bostock, Jason Davies, Will Turman, Jonathan Feinberg, and Kevin Chang, whose work inspired and helped create these visualizations.