As social networks such as Facebook and Myspace increase in popularity, so too does the complexity of these networks. Trends can be lost in the sheer size of the networks and numerical statistics can often fail to capture interesting relationships and patterns. Nevertheless, these trends and patterns are potentially valuable sources of information and efforts should be made to extract them.

By generating effective visualizations, complex relationships between users and variables can be represented in simple and transparent ways. Patterns which may have previously been hidden in numbers become obvious. The sheer mass of data is presented in an easily digestible format. The challenge of these visualizations is about discovering what data might elicit patterns, and how to represent this data so that patterns become salient.

Bertin[1] reveals 'visual variables' that can be used to take advantage of human sense of vision in order to make the properties of data more obvious. Tufte[3] explains principles of visualization as guidelines to creating a successful data representation. Donath[2] implements these strategies in a wonderful representation of social affiliations within a community by comparing users on dimensions relating to the newsgroups that they subscribe to using a spring based model.

Design Rationale

We felt that Facebook music tastes would be an interesting subject to model. When browsing Facebook through a web browser it is difficult to see which of your friends enjoy the same genres of music as you. Additionally, we predicted that patterns of clustering would emerge wherein people would be more likely to be friends with each other if they like the same kinds of music. We investigated the use of a spring-based technique, similar to what was used in Donath's system [2], except each user was an anchor instead of musical genres, so that we could compare people to each other.

Our visualization was a 2-D graph where each node represented a Facebook user and edges between nodes denoted friendships. We felt this was a straightforward and transparent way to represent a social network. Due to the limitations of Facebook, we only represented two main users and their friendships within the visualization.

We incorporated Bertin's principles of visual variables into the visualization design. For instance, the currently selected node in our visualization was enlarged, taking advantage of the selective property of size. User nodes were labeled a specific colour according to their taste in music. For example, rock fans were labeled blue. Since music genres have no inherent ordering or quantity, colour lent itself well to represent this data. We also used position to represent similarity. Users that had similar music tastes are located close together, while the inverse is true for dissimilar music tastes.

Similarly, we used Tufte's principles of information in our design. For instance, we included an overview panel where the entire network could be seen at all times. By doing this we provided a broad overview and fine detail. The user could hover over a node to receive additional information in the form of a tooltip, or click on a node to be shown even more information on a panel to the side.

We included a legend on the sidebar to specify the genre-to-colour mappings. We also felt that the legend gave the visualization some context and let the user know what exactly they were looking at.

Another feature we incorporated was a search bar. This allowed the user to quickly search for a specific Facebook user. Nodes which matched the search query were coloured bright red. Once again, this incorporated Bertin's principle of colour. Red is the most effective colour for creating a "pop-out" effect among low saturation items and we found that users could easily spot their search results.

Technical Description

Figure 1 describes the architecture of our system. It is a Pipe and Filter design which associates raw data from Wikipedia and Facebook and then transforms the data into a visualization using the algorithms described below.

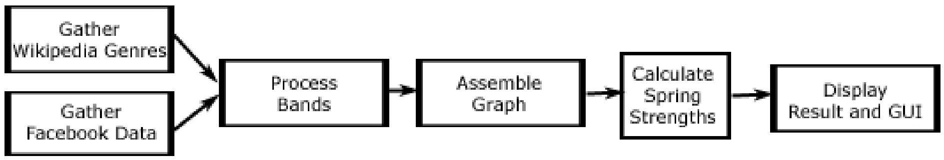


Figure 1. System Architecture

To mine the data needed for the visualization, we used both Facebook and Wikipedia. Wikipedia provided a list of band names and the genre that they belonged to. First, we went to the 'List of musicians' and assigned a genre to each of the links to lists found on that page. Then, we implemented a spider to go into each link and parse out the bands that were contained in each list. These were stored in a hash table. Then, to find the musical tastes for each friend on Facebook, we used the currently logged in user and got a list of their friends. All of the friend data was saved to a file in order to provide a fully populated and interesting demonstration.

For each friend, we parsed out a list of musicians from their 'Music' field and associated it to the hash table mined for the genres. We then associated a vector to each user that specified how many counts the user had for each genre. We then normalized the vector.

After that, we created a fully connected graph, where each user is connected by a spring to every other user. The cosine between each person was calculated, yielding a similarity value for their genre vectors. The strength of each spring was assigned to be proportional to the similarity between the vectors, effectively allowing more similar users to cluster together and less similar users to drift far apart. The distances are calculated only once since the high connectivity of the graph gives rise to high computational complexity.

Since our architecture is a one-shot Pipe and Filter, there is no visible output between the stages of operation – these stages serve to create data structures to be used by the clustering algorithm and GUI display.

Discussion and Conclusions

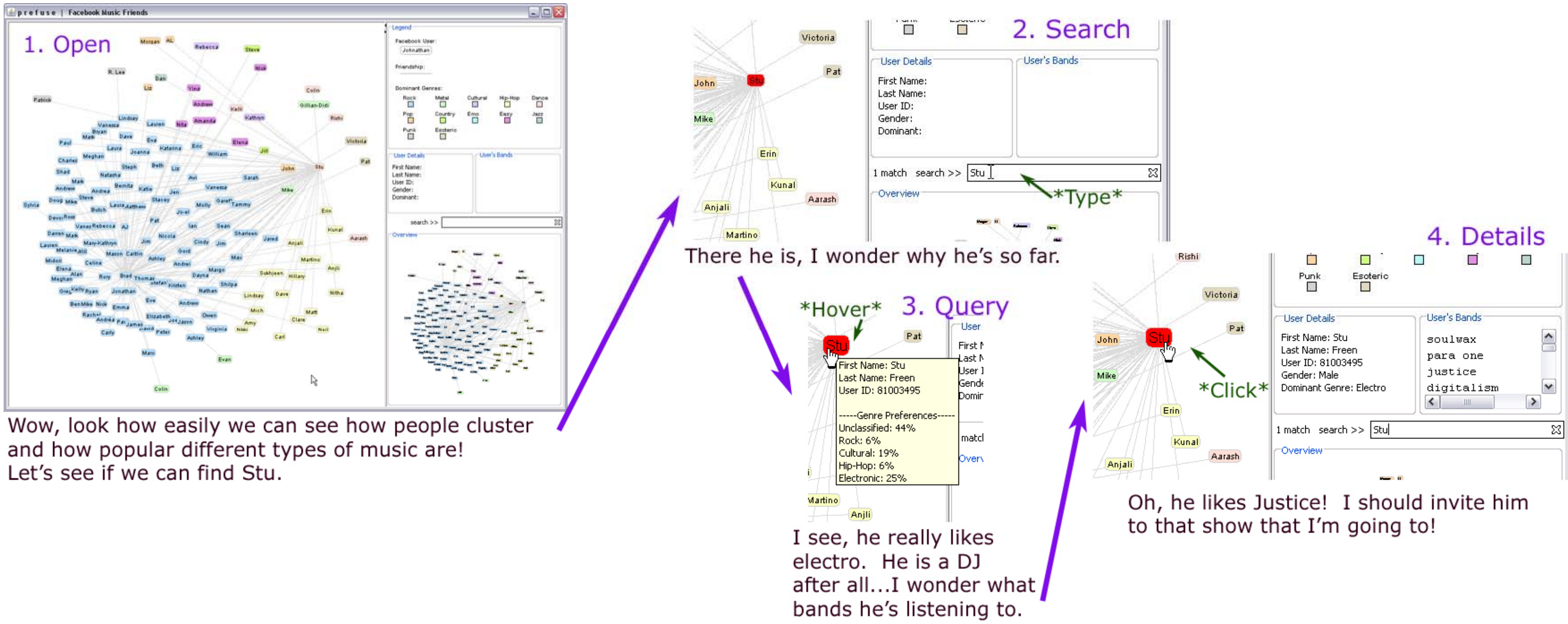
Appendix A describes a typical scenario, with screenshots, for using our visualization system. As shown in the scenario, the system facilitates the browsing of friends with similar music tastes by allowing the user to search specific friends, view their relative similarity, and on request, query details pertaining to the user and the reasoning behind the patterns shown.

Most systems in this domain attempt to visualize the mutual friendships between users. Our system visualizes more interesting and abstract information that a user could actually use if they were, for example, going to see a concert and wanted to invite friends.

One idea to improve our visualization would be to represent the genre counts as a pie graph instead of percentages inside of a tooltip. This would allow the distribution of data to be immediately transparent. We would also want to highlight listed bands by genre.

In an attempt to visualize interesting social networking data, we developed a system, using a spring-based technique to compare users based on their preferred genres. We believe that this system could be useful as a tool for finding more things in common with friends.

Appendix A Usage Scenario with Screenshots



Typical usage scenario and screenshots of visualization system.

References

[1] Bertin, J.(1983) *Semilogy of Graphics*. New and updated edition by Wisconsin Press (2003).
[2] Donath, J. *Visual Who*. In ACM Multimedia 95 November 5-9, 1995 San Francisco, California.
[3] Tufte, E. (2001) *The Visual Display of Quantitative Information, 2nd Edition.* Cheshire, Connecticut.