TagNavi – Interactive Tool for Tag Space Navigation

Jaeyoung Choi
University of California, Berkeley
jaeyoung@EECS.berkeley.edu

ABSTRACT
Tag cloud is a simple and widely used visual interface to browse tag space. However, the simple design of tag cloud limits its search capability. We describe an interactive tool to help navigate the tag space more effectively. This tool provides similarity based layout with interactive tag buttons and additional navigational aids to help user understand the tag space more easily.

Keywords
Visualization, Tagging, Navigation

INTRODUCTION
Tags are user-assigned, freely-chosen simple labels to annotate and categorize various kinds of resources on the web for future retrieval. Tagging is simple and it does not require a lot of thinking. People tag with one or more keywords to easily retrieve the resource in a later stage. In web tagging services like Delicious [5], Flickr [8], YouTube [20], LiveJournal [13], Last.fm [12], tagging has been successfully deployed to organize and share diverse online resources such as bookmarks, photographs, videos, blog posts, songs, and more. These tagging services provide users with a repository of tagged resources called tag space that can be searched and explored in different ways. Folksonomy is a system of classification derived from this collaboratively formed tag space.

In order to enable visual browsing of tag space, tagging services provide an interface model known as tag cloud. Tag cloud is a list of the most popular tags, usually displayed in alphabetical order, and visually weighted by font size. Unlike querying which requires user to formulate his information needs, this visual browsing interface allows user to recognize his information needs while scanning the interface.

Tag cloud is a simple and widely used visual interface model, but has some short-comings that limit its usefulness as a visual browsing interface. In a typical tag cloud implementation, tags are arranged alphabetically as shown in Figure 1. However, alphabetical arrangement does not facilitate visual scanning of tag cloud nor enable to draw semantic relations between tags. Folksonomy is defined as a flat space of keywords without previously defined semantic relationships between tags. However, Brooks [3] shows that associative and hierarchical relationships of similarity between tags can be obtained from the tag co-occurrence analysis. Also Begelman [1] suggested that clustering based on co-occurrence can be used to group related tags together.

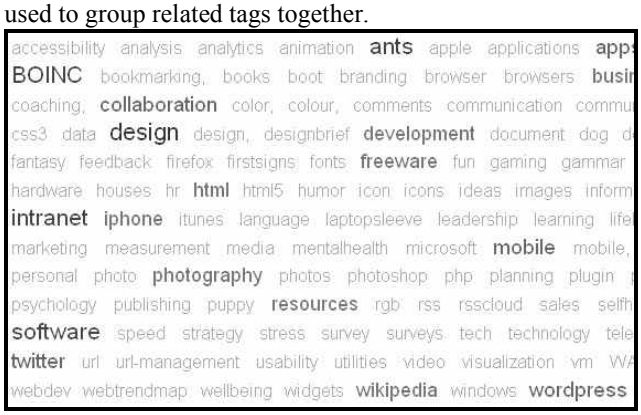

Figure 1. Tag Cloud from Delicious.

To meet the space constraint and to avoid being a cluttered list of tags, only the subset of whole tags can be shown at a time. Usually, this selection of tags to display is based exclusively on the use frequency. This leads displayed tags to inevitably have a high semantic density [18]. Begelman [1] also states that very few different topics with all their related tags tend to dominate the whole cloud. Thus, to improve tag clouds, a new tag selection method is needed.

Typical tag clouds don't allow user to select time range and thus they only represent the overall trend of tag space. Flickr shows tags that were the most popular for the last 24 hours and last 7 days [9] , but the date range is fixed. This seriously limits the potential of navigating tag space as user cannot get an answer to the questions involving time such as, "Which tag was the most popular during last Presidential election period?", or "When was the tag 'volcano' used the most?"

In this paper we describe an interactive visualization method for more effective display of tags along with some additional navigational aids to extend the area of search. We first review related work in the area of tag cloud visualization.

RELATED WORK
HTML-based tag clouds have been appeared on numerous web sites such as Delicious [5], Flickr [8], YouTube [20],

and so on. Tag clouds on these web sites are in the simplest form by alphabetically arranging tags and deciding font size based on their use frequency. Some sites use additional color schemes to encode the tag use frequency.

Kaser [11] presents models to improve the display of tag clouds that consists of in-line HTML by utilizing typesetting and rectangle packing. Millen et al.[14] states a tag selection method by proposing that user be dynamically able to remove the less significant tags. Bielenberg [2] has proposed circular clouds, where the most heavily weighted tags appear closer to the cluster.

There are several approaches to represent tag space outside the conventional form of tag cloud. Dubinko et al [6] have proposed a model to represent tags over a time line. Jaffe et al. [9] proposed a tag cloud integrated with maps for tags having geographical information, such as pictures taken at a given location.

There are many prototypes of interactive visualization of tag space on the web. HubLog [10] shows a simple tree of related tags exclusively based on their co-occurrence rate. SpaceNav [17] shows relation between tags by placing user selected tag in the center and placing its related tags around in circle. Opaque circle surrounding the related tags represent the weight of co-occurrence with its size of area. Stefaner's Elastic Tag Map [7] proposes an interactive tag map which uses PCA algorithm for the layout of tags.

MATERIALS

We built an ad-hoc programmed crawler with Ruby programming language to scrape off the data set we used in this project. Data sets were collected from social bookmarking website Delicious [5]. Delicious allows free-for-all tagging (user can freely choose one or more tags to annotate the resource) and blind-tagging (user does not see which tags were used by users to annotate the same resource). Each bookmark record consists of the following sets of data : URL of the bookmark, name of the bookmarked page, date of bookmark, user name, tags.

METHODS OF LAYOUT

Tag Similarity

The simplest way to define similarity between two tags would be to use co-occurrence of two tags, which is the number of times that two tags are assigned to the same resource. Cattuto et al. [4] states that some semantic relationship can be found from the co-occurrence of tags. Begelman [1] introduces the concept of strongly related tag for the representation of relationship between tags, based on the raw co-occurrence of two tags.

However, observation shows that extremely popular tags have high co-occurrence values with many other weakly related tags [15] and using the raw co-occurrence causes bias. To avoid this, we calculate the normalized co-occurrence (NCO) by dividing the raw co-occurrence value by the popularity of two tags using Jaccard Index [1]. Other normalizations such as cosine similarity are also possible.

$$NCO = \frac{|A \cap B|}{|A \cup B|}$$

Here A is the set of documents tagged with tag a, and B is the set of documents tagged with tag b.

Clustering Algorithm

Alphabetical based layout scheme may be convenient when user knows what tag he is looking for such as when browsing his own tag space. However, when navigating a previously unknown tag cloud, it could become a chaotic list of tags. To improve this, we used clustering algorithm to provide similarity based layout. We tested two data clustering algorithms for the layout.

The first one was K-means clustering algorithm. Although there is a question of deciding which value to use for K, K-means converges reasonably fast and does show improved layout when compared to alphabetically arranged layout as similar tags are grouped together.

The second algorithm was the hierarchical divisive clustering algorithm presented in Simpson [15]. It starts from a similarity graph, where nodes represent tags and edge weights represent the similarity. We briefly describe the algorithm we used.

Starting with a graph G which contains all tag relationships :

1. Count the number of clusters present in G by counting the disconnected sub-graphs

2. Evaluate the current clustering using a quality measure called *modularity* [15]. If *modularity* > all previous modularities, set $G_{maxmod} = G$.

3. Remove the edge with the lowest NCO value from G.

4. Repeat process from step 1 until no edges are left. Heuristic was used to reduce the number of iterations by stopping when modularity exceeds a certain threshold.

Hierarchical divisive algorithm has its advantage that we do not need to specify the number of semantically related clusters as we do with K-means clustering algorithm. This is a great advantage since every tag space has different number of main topics. However, hierarchical divisive algorithm's running time depended heavily on the value of threshold to stop the algorithm, and the threshold which produces reasonable clusters usually took too much time (as high as 10 seconds). This is critical as we need this algorithm to run every time a user selects a new date range.

Since our layout displays the list of each tag cluster into columned group, number of clusters did not matter much with K-means algorithm. Currently we are using K-means algorithm as its fast speed is suited to be used in an interactive environment. We are using the value 8 for K.

Figure 2. Snapshot of TagNavi. From the top, date range is set between May 2008 and Feb 2010. User can observe the overall tag use history from the topmost history graph (colored red) along with each of selected tag's history to the bottom. The bottommost history graph shows the history of when selected tags were used altogether. Tags that are related to both 'media' and 'cooperacion' retain their size while all other tags were minimized. Since there are only 1 or 3 resources annotated with the related tags, frequency graph located at the bottom-left side shows very little bar.

Tag Selections

In many cases, even single user's tag space contains a large set of tags that they cannot all be displayed. Displaying all tags does not necessarily convey more information to the user as tag cloud will be cluttered with too many information that it loses effectiveness as a visual browsing tool. Thus, we need to decide which tags should be selected and which is to be cut off.

Tag selection was not handled much in detail in the current design. Instead of cutting off the tags based on their use frequency in its entire tag space, using the clustering algorithm first and then cutting off the least used tags from each cluster seemed to do the job.

DESCRIPTION OF INTERFACE

Distribution of Font Sizes - Logarithmic distribution of size scheme

Distribution of font size in a tag cloud may seem trivial, but in fact, it is very important. The size of a tag represents the relative importance of the tag in the whole tag cloud. If we fail to properly represent that, user will be given an incorrect impression of the relative weight.

Most implementations of tag cloud use the linearly distributed size scheme. They compute the range of use frequency by taking the differences between the maximum use frequency and the minimum use frequency in the cloud. This value is divided by the number of bins (usually 4 to 5) to retrieve the distribution. Since the distribution of use frequency form a long tail distribution, and not evenly

spread across the range of use frequency, user will see a few largest-sized tags with an extremely high weight, and large number of smallest-sized tags with very low weight.

For our tool, instead of using the raw use frequency value, we use the log value to assign the bins. Using logarithmic scale gave us a more evenly distributed size of tags.

Date Range Selector

Date range selection is one of most basic search options. However, it wasn't incorporated into any of previous tag-cloud tools. Ability to select date range gives more search option to the user. Also, it enables user to observe the change of tag trend. Moreover, user can explore further by looking at the selected tag's overall use history to see the trend of each or total combination of tags.

Tag Usage History Graph

This graph allows user to see the history of tag use frequency. User can select one tag to observe its history, or select more than one tag to observe each of their history along with the history which combination of selected tags were used together. Brightness was used to encode the relative frequency of each bar as many users tend to use same tag heavily on a single day. Figure 2 shows a snapshot of TagNavi which illustrates the usefulness of this feature.

Tag Buttons

Tag buttons respond to the user action by highlighting or changing their size. When the mouse is over a tag, all other tags which have co-occurrence are slightly enlarged (30% increase to original size) to help scan through the related tags. User can click on a tag to lock the selection. If this happens, only the tags which co-occurred with the selected tag retain their size and the other irrelevant tags reduce their size significantly. User can click on other remaining tags to make multiple selections and only the tags which co-occurred with the combination of selected tags remain. As shown in Figure 3, during this process, number of resources associated with each of remaining tags will be shown next to each tag. This feature is aimed to help the user to easily narrow down the category by choosing related tags.

Frequency graph

It's reasonable to connect broader tags (with high use frequency) to more general purpose search tasks, and narrow tags (with low use frequency) to more specific and goal-oriented search tasks. Based on this idea, we introduced frequency graph to aid the search.

Since font size merely divides the tags into large groups based on their relative weight, frequency graph provides user with the information of actual weight distribution of tags. Basically, frequency graph is a bar graph showing the weight of each tag sorted left to right from the highest to the lowest. As shown in Figure 4, frequency graph gives user the overall tag usage pattern of entire tag space. Each bar

Figure 3. A portion of tag buttons when tags 'education' and 'elearning' are selected. Tag 'social' shows a number in parenthesis to the left. This represents the number of resources associated with the selected two tags and itself.

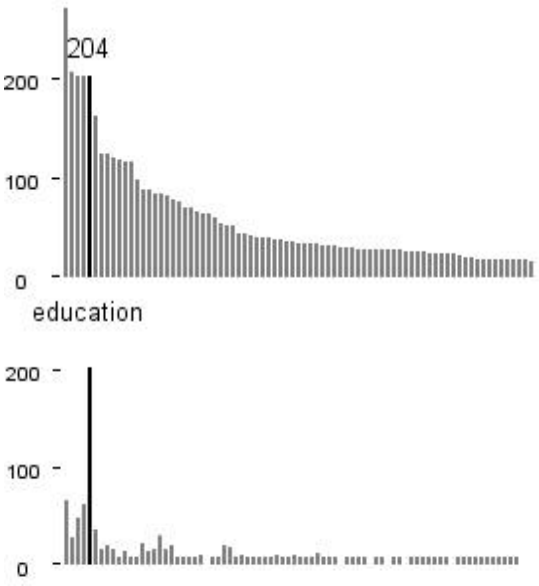

Figure 4. (Above) Frequency graph when mouse is over the tag 'education'. (Below) Frequency graph when tag 'education' is clicked on. Only the bars that correspond to the tags which co-occur with 'education' remain. Their length is changed to represent the frequency of co-occurrence.

corresponds to the tag shown in the tag cloud, and when the mouse is put over the bar, corresponding tag highlights. Likewise, when mouse is over the tag, corresponding bar highlights. This proves to be useful, for example, when user move mouse over the tail part of the frequency graph to traverse through the tags that have high discriminating factor.

Observing the shape of frequency graph itself may give rough information about the pattern of the user's tagging behavior. When a few tags dominates the tag space (for example, 'cool' or 'web' which appear with almost any bookmarks in many user's tag space), frequency graph shows steep left peak. These tags don't have any discriminating factors, and tagger can be considered to be not thoughtful when selecting keyword. On the other hand, if the left part of frequency graph has a slow curve, we can tell that the tagger used more careful selection of keywords to categorize the resources.

If one or more tags are locked, only the related tags remain highlighted and the frequency graph shows the same effect as well. Bars that correspond to the related tags would still

show, but their length is adjusted to the number of co-occurrence with the locked tags. Ordering of the bar is remained the same to encode both the co-occurring frequency and its own use frequency in the whole tag space.

DISCUSSION

The new visualization of tag cloud gives more search options to the user which conventional tag clouds do not provide. Ability to select date range and the presence of history graph that shows history of tag usage greatly improve browsing experience. Similarity based layout scheme helps user to understand the overall trend of tag space better, but more optimization and heuristics should be added to the current clustering algorithm. Depending on the tag sample, the algorithm sometimes create clusters which are not reasonably divided or sometimes even fails to converge in a reasonable time when the number of tag is small. Selection of tags after clustering seem to have more discrimination value as more tags with less use frequency is shown. However, we will need a formal measure to verify this in the future. Size-changing interactive tag buttons allowed user to easily search the tag space based on the co-occurrence of tags.

Displaying each cluster as one column led to large amount of whitespace between tags. One of the feedbacks we received stated that, he got an impression that there still are too many tags displayed. Some other mentioned that current grid-like layout helps understanding the overall trend better when compared to tighter layout. We will need further improvements and evaluations on the layout design.

FUTURE WORK

More work on preprocessing of tags is required to integrate singular/plural terms and synonymous terms. We believe that this will give us better results with the clustering as integration of related tags will lead to stronger relationship with other tags.

We will need to provide the layout customizability for the small screen or mobile environment. Current version will not be useful in small screen or mobile interface as it will involve too much scrolling. Also, making a web version of the current application and designing the layout to be integrated into existing blogs and sites should follow.

ACKNOWLEDGEMENT

We would like to thank Maneesh Agrawala for giving us an informative and interesting lecture. We also would like to thank many friends who gave us thoughtful feedbacks on the design of this tool.

REFERENCES

1. Begelman, G.; Keller, P. and Smadja, F. (2006). Automated Tag Clustering: Improving search and exploration in the tag space. WWW2006
2. Bielenberg, K. (2005) Groups in social software: Utilizing tagging to integrate individual contexts for social navigation. Master's thesis, Universitat Bremen.
3. Brooks, C.H. and Montanez, N. (2006) Improved Annotation of the Blogosphere via Autotagging and Hierarchical Clustering. WWW 2006
4. Cattuto, C.; Loreto, V. and Pietronero, L. (2006). Collaborative Taggin and Semiotic Dynamics. Unpublished paper
5. Delicious. http://delicious.com/.
6. Dubinko, M.; Kumar, R.; Magnani, J.; Novak, P. Raghavan and Tomkins, A. Visualizing tags over time. In 15th International World Wide Web Conference, pages 193-202.
7. Elastic Tag Map. http://well-formed-data.net/experiments/tag_maps_v5/
8. Flickr. http://www.flickr.com/.
9. Flickr Popular Tags http://www.flickr.com/photos/tags/
10. HubLog.http://hublog.hubmed.org/archives/001049.html
11. Kaser, O. and Lemire, D. (2007) Tag-Cloud Rawing: Algorithms for Cloud Visualization. WWW2007
12. Last.fm.http://www.last.fm/
13. LiveJournal. http://www.livejournal.com/
14. Millen, D. R.; Feinber, J. and Kerr, B. : Social Bookmarking in the enterprise. In CHI '06
15. Simpson, Edwin. Clustering Tags in Enterprise and Web Folksonomies (2008)
16. Association for the Advancedment of Artificial Intelligence
17. SpaceNav. http://www.ivy.fr/revealicious/demo/spacenav.html
18. Y. Hassan-Montero and V. Herrero-Solana. Improving tag-clouds as visual information retrieval interfaces. In
19. International Conference on Multidisciplinary Information Sciences and Technologies (InSciT2006).
20. Youtube. http://www.youtube.com/