Tomographic Clustering To Visualize Blog Communities as Mountain Views

Belle L. Tseng
NEC Laboratories America
10080 N. Wolfe Road, SW3-350
Cupertino, CA 95014 USA
+1-408-863-6008
belle@sv.nec-labs.com

Junichi Tatemura
NEC Laboratories America
10080 N. Wolfe Road, SW3-350
Cupertino, CA 95014 USA
+1-408-863-6021
tatemura@sv.nec-labs.com

Yi Wu
University of California, Santa Barbara
Dept. of Elec. and Comp. Engineering
Santa Barbara, CA 93106
+1-805-893-7788
wuyi@ece.ucsb.edu

ABSTRACT
Blogs have created a fast growing social network on the Internet. However ranking solutions are not sufficient to capture relationships between important blogs and between communities. In our paper, we combine blog rankings with their social connections to provide a framework to understand multiple blog communities. A novel mountain view visualization is provided to explore different communities of interest in blogspace. The mountain views are generated using a tomographic clustering algorithm on the blog social network. The mountain view shows mountains of communities consisting of connected blogs. Peaks and valleys of the mountain view depict representative blogs as community authorities and community connectors, respectively. We developed a retrieval and exploratory system to illustrate this framework, and perform initial experiments to validate the results.

Categories and Subject Descriptors
H.3.3 [Information Systems]: Information Search and Retrieval; H.3.5 [Information Systems]: Online Information Services; H.5.4 [Information Systems]: Hypertext/Hypermedia

Keywords
Blogs, community, information dissemination, influence, authority, ranking, clustering, graph connectivity.

1. INTRODUCTION
Recently, blogs (or weblogs) have become prominent social media on the Internet that enable users to quickly and easily publish content including highly personal thoughts. A blog is typically a web site that consists of dated entries in reverse chronological order written and maintained by a user (blogger) using a specialized tool. Since a blog entry can have hyperlinks to web pages or other blog entries, the information structure of blogs and links (sometimes called the blogspace) can be seen as a network of multiple communities.

By visualizing the blogspace as a social network of bloggers, various techniques provided by the social network research community can be introduced, for example, how blogs communicate with each other. In addition to traditional analyses, however, we must consider special characteristics of the blogspace compared to typical social networks in the real world: dynamics and divergence of popularity and quality of blogs.

WWW 2005, May 10--14, 2005, Chiba, Japan.

Although there is no predefined hierarchical structure in the blogspace (unlike mass media), a blog community is not "flat". Compared to the real world network, one can become much more easily known to others in the blogspace. For instance, a blog that provides interesting entries can attract worldwide audiences and get citation links from others. In this manner, a "celebrity" within an interest community can emerge dynamically. On the other hand, since anyone can publish and link to any blog, there are many unimportant or even harmful entries such as spam. Link-based rankings such as PageRank [9] are useful to detect popular blogs. To analyze community structure, however, a simple ranked list does not provide enough information. Instead popularity (given by ranking score) is incorporated into the connections between blogs, and we introduce the concept of "ranking-based connectivity."

Our focus is not on the entire network of blogs but on a social network of important blogs (i.e., blogs taking important roles in a community and trusted by the community members). Useful and trusted discussions can be extracted from such a network. If two "celebrities" have a common topic of interest, it is natural that they are aware of each other and likely to start communication (publishing entries referring to each other) and produce valuable discussions referred to by the community audiences. This structure of a blog community can emerge everywhere in the blogspace. Even in a particular topic, there can be multiple diverse communities. Our goal is to capture the community landscape on a specific topic and allow users to explore these important blog communities.

In this paper, we propose (1) "mountain view," a new visualization technique that provides a landscape of blog communities in terms of popularity and connectivity, (2) "tomographic clustering," the underlying algorithm that generates a mountain view from a social network of blogs with ranking score information, and (3) architecture for community retrieval and exploration system based on the algorithm and visualization.

Our system allows the user to specify a query and retrieve a mountain view of the topic, through which the user can explore various communities. The paper is organized as follows. Section 2 reviews related work on blogspace and community extraction. In Section 3, we propose our system and algorithm to help a user understand the community structure of blogs on a specified topic. Experimental setup for data collection and preliminary results on the data are shown in Section 4 and Section 5, respectively. Section 6 summarizes our conclusions and future works.

2. RELATED WORK

Recent research on the blogspace focuses on two major aspects: temporal analysis and information diffusion among blogs. For temporal, Kumar proposed a method to identify bursty community of blogs based on community extraction and burst analysis [6]. Adar introduced the Epidemic Profile to characterize the temporal behavior of posting entries that refer to specific URLs [1]. BlogPulse is a system that provides trend graphs that shows the popularity of specific topics over time [3]. Gruhl analyzed temporal characteristics of blog postings on a specific topic [4].

To study information diffusion, Adar proposed a method to find implicit information propagated between blog entries and presented iRank, a ranking algorithm based on the implicit link structure [1]. Gruhl introduced a probabilistic model of information propagation among individuals and proposed an algorithm to induce a transmission graph that captures information diffusion structure in the blogspace [4]. Our focus is, given the explicit blog influence as a network, to understand the underlying community structures. Our framework incorporates various results from graph-based information diffusion.

There have also been extensive studies done on identifying "communities" from the link structure of the Web. The HITS algorithm identifies "hub" and "authority" web pages, where a hub links to many authorities and an authority is linked by many hubs [5]. In this algorithm, a community is modeled as a bipartite graph of hubs and authorities. The trawling algorithm [7] has been proposed to discover such communities (i.e., dense bipartite subgraphs) from a huge data set. Flake defines a community on the web as a set of sites that have more links to members of the community than to non-members, and models the graph partition based on maximum flow and minimal cuts [2]. The community chart algorithm [10] identifies multiple communities and relationships between communities. Communities are identified by partitioning Symmetric Derivation Graph whose link represents relationship between two sites such that each site is derived as a top-N authority from the other site. In this paper, our focus is not on extracting subgraphs of general link structure but on understanding the graph structure of ranked nodes.

3. COMMUNITY UNDERSTANDING

A key challenge to explore the blogspace is to understand the different blog communities and identify important representatives. In this section, our objective is to allow users to interactively understand the blogspace by providing a system framework for retrieving relevant communities and interactively explore the authoritative and allied blogs.

3.1 System Overview

The blogspace is growing and becoming more difficult to identify important blogs for a user's topic of interest. We propose a system framework that will allow a user to specify a query, retrieve relevant communities, and interactively explore the communities by examining the extracted representative blogs. To achieve this objective, our system is composed of two major components, (1) Community Retrieval and (2) Community Exploration. Figure 1 illustrates the two modules.

In Community Retrieval, the module allows the user to specify a topic of interest and returns the highly-relevant clustering of communities, which we refer to as the *Mountain View* of this query. When a query is given, the Entry Retrieval component retrieves relevant entries from the blogspace. These relevant entries allow the Blog Ranking component to rank the blogs, as will be described in the following subsection 3.2. Following, Tomographic Clustering is performed to discover community clusters of connected and relevant blogs, as explained in subsection 3.3. The clustering result generates a mountain view representation of the discovered communities corresponding to the user query.

In Community Exploration, the module takes the returned mountain view of a query and provides the user with a visual exploration of the relevant communities and representative blogs. The user can get a high-level view (Alta Vista) of the retrieved communities in this query space by observing the visualization. In addition, the user may be interested in examining the top-ranked community or alliances between communities. Furthermore, representative blogs can be extracted to represent the complex community landscape.

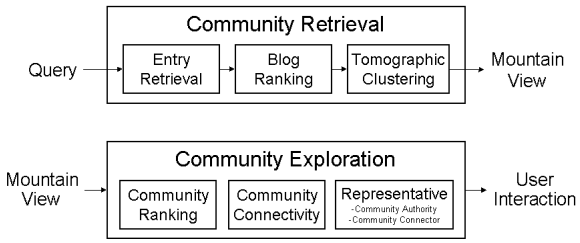

Figure 1. System Overview of Community Retrieval and Community Exploration to Understand Blog Communities.

In our retrieval and exploration system, the user specifies a query and interactively examines interesting communities with various levels of details via the mountain view representation. One application includes a blog topic summarizer, where the user identifies a topic and the system shows a list of representative blogs extracted from each of the top-N ranked communities. This allows the user to quickly get an overview of the different communities in this topic. Furthermore, our system provides a framework for other applications to be build based on the derived mountain view representation.

3.2 Blog Ranking

To locate valuable web contents on the Internet, search engines are used to retrieve a rank ordering of important pages with respect to the user's request. Similarly with web log contents, blogs can be retrieved and ranked. Our goal in this subsection is to illustrate one method to rank the blogs.

When a user identifies a query of interest, the retrieval module needs to find relevant entries before blogs can be ranked. The steps are as follows. First, relevant entries are identified. In our system, an entry is labeled relevant with respect to the query if the query term exists in the entry. Second, the impact scores of relevant entries are calculated. Finally, the ranking scores for the blogs are derived from the entry scores.

The impact of entries may be calculated by an iterative procedure much similar to the PageRank algorithm used in the Google search engine. Let entry graph *EG* be denoted by the set of entries $E = \{e_i\}$ and set of entry links $EL = \{(e_i, e_j)$ *where entry* e_i *cites entry* $e_j\}$, thus $EG = (E, EL)$. Given an entry graph *EG*, the score of an entry $s(e)$ can be calculated by the following iterative Equation 1.

$$s(e_i) = (1-d) + d\sum_{e_j \in IN(e_i)} \frac{s(e_j)}{|OUT(e_j)|} \qquad \text{Eq. (1)}$$

Here $IN(e_i)$ represents the set of entries that cites e_i, $IN(e_i) = \{e_j \mid (e_j, e_i) \in EL\}$, and $OUT(e_i)$ represents the set of entries that e_i cites, $OUT(e_i) = \{e_j \mid (e_i, e_j) \in EL\}$. Consequently, $|OUT(e_i)|$ represents the total number of entries that e_i cites. Finally, d is a parameter to control the damping factor to the rank propagation.

Next we consider a set of blogs $B = \{b_i\}$. Each blog b_i owns a set of entries E_i. There can be multiple entry-to-entry links from blog b_i to blog b_j, which we denote as entry link $EL_{ij} = \{(e_k, e_l) \mid e_k \in E_i, e_l \in E_j, (e_k, e_l) \in EL\}$. Thus the social network of blogs B can be represented as a graph $G = (B, L)$ where the set of links $L = \{(b_i, b_j) \mid EL_{ij} \neq \phi\}$.

There are various ways to score blogs based on the entry scores. Currently, we take the following approach to calculate the ranking score $s(b)$ of blog b, as shown below in Equation (2).

$$s(b_i) = \sum_{b_j \in IN(b_i)} \left\{ \frac{1}{|EL_{ij}|} \sum_{(e_k, e_l) \in EL_{ij}} s(e_k) \right\} \qquad \text{Eq. (2)}$$

Here $IN(b_i)$ represents the set of blogs that have entries citing to blog b_i, $IN(b_i) = \{b_j \mid (b_j, b_i) \in L\}$. We incorporate the normalization $|EL_{ij}|$ to refer to the total number of entry links from blog b_i to blog b_j. As a note, we have tried other blog ranking methods like Equations (3) and (4),

$$s(b_i) = \frac{1}{|E_i|} \sum_{e_j \in E_i} s(e_j) \qquad \text{Eq. (3)}$$

where the blog score is the average of the blog's entry scores, and

$$s(b_i) = \sum_{e_j \in IN(b_i)} s(e_j) \qquad \text{Eq. (4)}$$

where the blog score is the sum of scores from citation entries. However these do not resolve certain deficiency that we observed.

There are two underlying reasons we chose Equation (2) to calculate our blog ranking scores. First, we observed that there was a high variance in the total number of entries that a blog can own, variance of $|E_i|$ is large. We find that some blogs have a large number of entries and most of these entries are not referred to by other blogs. In order to focus on blogs that have most entries referred to by other blogs, we needed to derive a score that accumulates the effect of references by other blogs.

Second, we observed that there was a high variance in the total number of links between blog b_i and blog b_j, variance of $|EL_{ij}|$ is large. Some blogs have a very large number of links between each other (e.g., multiple blogs owned by a single person). Instead of accumulating the effect of each reference between two blogs, we chose the average of those. As a result, Equation (2) seems to capture our desired outcome.

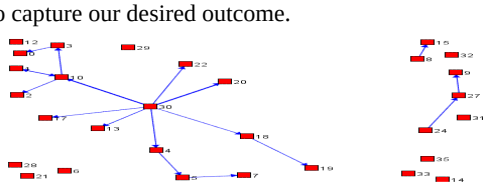

Figure 2. Social Network of Top 30 Blogs for "google" query.

For our example query, we chose the topic "google". We perform blog retrieval and ranking using Equation (1) for ranking the entries, followed by Equation (2) for ranking the blogs. Figure 2 displays the results of the top ranking 30 blogs by the red squares and the blog citation links between them, subsequently referred to as the blog social network. The numbers depicted next to the blogs denote the blog IDs used for cross-referencing. Also, the heavier the blog links, the stronger their relationships are. As evident from the figure showing the "google" blog social network, there is one large strongly-linked community and some smaller and less-connected communities. In the following subsection, we explore how to discover and compare these communities.

3.3 Community Discovery

3.3.1 Ranking-Based Connectivity Analysis

The social network in Figure 2 shows multiple disconnected communities of top-ranked blogs. However note that the connectivity of these graphs depends on the threshold of ranking results – if another blog is added by lowering the threshold, it may connect multiple communities together. In fact, the ranking of blogs is important when we consider social connectivity of blogs. Blogs linked by a highly-ranked blog appears more connected than blogs linked by a lower-ranked blog (for example, the lower ranking blog may be a spam). We call this connectivity as "ranking-based connectivity."

A formal definition of ranking-based connectivity is given as follows. Let $G = (B, L)$ be a graph that represents a social network of blogs, where B is a set of blogs (vertices) and $L = \{(b_i, b_j) \mid b_i, b_j \in B\}$ is a set of social connections (edges). G can be represented as a disjoint union of connected subgraphs $G = \bigcup C_i$, where C_i represents a connected subgraph. Let B_t be the top-ranking set of blogs with threshold t (i.e., $B_t = \{b \mid b \in B, s(b) \geq t\}$) and G_t be the induced subgraph of G corresponding to B_t (i.e., the graph after removing blogs whose scores are under the threshold t). Then given a threshold t, a set of connected subgraph $\{C_i^t\}$ is given as $G_t = \bigcup C_i^t$. The rank-based connectivity rc(b_1, b_2) between two blogs b_1 and b_2 is defined as the maximum threshold t such that $\exists i : b_1, b_2 \in C_i^t$. The critical blog that connects multiple graphs into C_i^t is called the connecting blog of b_1 and b_2, denoted by $bc(b_1, b_2)$.[1] The score of the connecting blog is equal to the rank-based connectivity, s($bc(b_1, b_2)$) = rc(b_1, b_2). Ranking-based connectivity between two disjoint connected-subgraphs (i.e., two blog communities) is defined similarly.

3.3.2 Tomographic Clustering

We are interested in reflecting the changing structure of $\{C_i^t\}$ when t varies from the minimum score to the maximum score. Although a snapshot can be seen in Figure 2, we want to capture a concise representation that shows the community structure given the rank-based connectivity analysis.

We propose an algorithm that provides such a representation, called "tomographic clustering." Tomography is taken from computer tomography (CT) scan that generates 3D structure of a human body by accumulating 2D "slices." Given different

[1] There may be multiple connecting blogs with an equal score.

thresholds of ranking score, different "slices" of the community structure is given as a set of clusters. The overall structure can be understood through accumulation of multiple slices.

Tomographic clustering generates an ordered sequence of blogs $s = (b_1,\ldots,b_n)$ that represents the ranking-based connectivity of blogs in the following manner: When the sequence s is split into a set of contiguous subsequences $\{s_i^t\}$ by removing blogs under the threshold t, the set of blogs in each subsequence s_i^t is equal to the set of blogs in C_i^t.

Figure 3 illustrates a visual interpretation of the clustering result. A curve called the mountain view is drawn by plotting blogs in the order of the sequence s with their scores on the vertical axis. If the mountain view is cut at the score threshold t, the set of curves above t corresponds to the set of connected subgraphs $\{C_i^t\}$.

Mountain view can be seen as the "contour" of the community structure since the curve shows upper bounds of paths within the social network. For instance, suppose there are blogs b_1, b_2, b_3 in this sequential order, and $s(b_2)$ is lower than $s(b_1)$ and $s(b_3)$, then any path between b_1 and b_3 must contain b_2 or other blogs with a score lower than b_2.

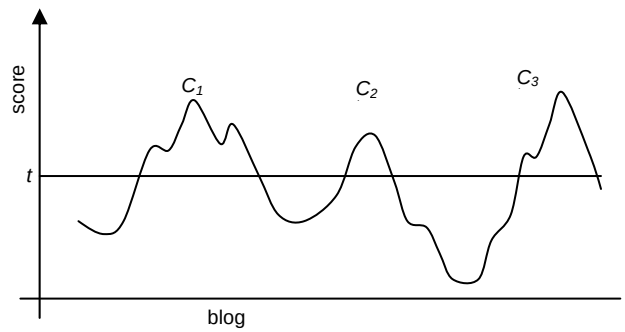

Figure 3. Mountain view visualization to illustrate multiple communities of interest.

3.3.3 Mountain View

In the mountain view, any top portion of the mountain cut by a horizontal line can be seen as a community. In Figure 3, the threshold t cuts the mountain view into 3 such communities, C_1, C_2, and C_3. Blogs within each community are connected by a ranking-based connectivity value higher than the threshold t.

Given a community as a part of the mountains, two types of representative blogs can be visually identified in the mountain view.

(1) Community authority $a(C)$: The blog with the highest score in the community is visualized as the highest mountain peak.
(2) Community connector $c(C_i,C_j)$: The connecting blog between communities C_i and C_j is visualized as the lowest point in the valley between the C_i and C_j communities.

In the next subsection, we will describe the tomographic clustering algorithm to identify these two types of representatives.

3.3.4 Tomographic Clustering Algorithm

A sequence of blogs that satisfies the definition of tomographic clustering exists if duplication of a blog is allowed in the sequence. A sequence can be generated by tracing the clustering process of $\{C_i^t\}$ from the maximum score to the minimum score: instead of merging sets of nodes, the algorithm concatenates sequences.

```
cluster() returns sequence {
  Let sequence_set S = 0
  For each blog b in B in the ranking order
    Let sequence_set Sc =
                    {s | connects(b,s), s in S}
    If (Sc = 0) S = S + {(b)}
    If (|Sc| = 1) let sequence s in Sc,
                 S = S – Sc + {add(s,b)}
    If (|Sc| > 1) S = S – Sc + {concat(Sc,b)}
 Return concat(S, b0)
}
```

Starting from the top rank blog, the algorithm adds blogs in the ranking order and maintains a set of sequences as an intermediary result. For each blog b added, the algorithm checks whether b is connected with any of the current sequences. If it is not connected with any sequence, a new sequence that contains only b is created. If it is connected with only one sequence, b is added to that sequence. If it is connected with multiple sequences, they are concatenated into one sequence with insertion of b between two sequences. Blog *b0* is an imaginary blog that connects all blogs in B, which is introduced to generate one sequence as the final result.

connects(b,s) returns a boolean value that represents whether the blog b is directly connected to any blog in the sequence s.

```
connects(b:blog, s:sequence) returns boolean {
    If (exists b' in S such that
         (b',b) in L or (b',b) in L)
    return true
    Else return false
}
```

add(s,b) creates a sequence by adding blog b to sequence s at either the head or tail. In this algorithm, the head and tail is chosen in alternatively for esthetic reasons when the sequence is visualized.

```
add(s:sequence, b:blog) returns sequence {
    If (|s| is odd) return s + (b)
    Else return (b) + s
}
```

concat(S,b) creates a sequence by concatenating sequences in a sequence set S with inserting blog b between sequences. Sequences can be concatenated in any arbitrary order.

```
concat(S:sequence_set, b:blog) returns sequence
{
    If (|S| = 1) return s where s in S
    Else return s + (b) + concat(S – {s}, b)
          where s in S
}
```

3.4 Community Exploration

In this section we will provide a visual exploration of communities using the mountain view. Figure 4 illustrates how the user can explore community structure visualized as a mountain view. The user can extract a community as a portion of the mountain by specifying "a cutting line". Given this community as the focused community, related communities can be extracted as a set of maximal communities connected to the focused community. In Figure 4, the user specifies the community in focus C_0 and acquires, as a result, a set of related communities $C_1, \ldots, C_6$. The user can get a rough idea on the importance and relevance (connectivity) of a related community (C_i) from the height of the local peak $s(a_i)$, the depth of the valley $s(c_i)$, and the size of the mountain.

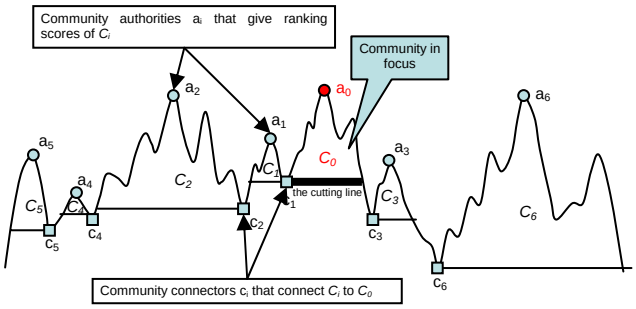

Figure 4. Mountain view of multiple communities showing community authorities and allied connectors.

In addition, the system may provide a summary of content information for each community (such as keywords). The granularity of a related community is given based on its rank-based connectivity with the focal community C_0 (i.e., the weaker the connection, the coarser the community). This feature provides the focal information with contextual information to navigate the user to various communities. For example, C_6 is given as a coarse-grained cluster with respect to C_0 because it is weakly connected (i.e., the rank-based connectivity $rc(C_6, C_0) = s(c_6)$ is low). If the user wants to focus on more detailed community structure within the community C_6, he or she can cut out a sub-community from C_6. By changing the focal point (i.e., C_0) in this manner, the user can explore multiple communities in the mountain view.

4. EXPERIMENTAL SETUP

For our initial experiments, we have developed a focused crawler to collect blog data from the Internet. The crawling process is divided into five components, (1) initial seeds, (2) blog discovery, (3) entry extraction, (4) seed expansion, and (5)keyword retrieval.

For initial seeds, we start with a focused list of topic keywords. Given a keyword, the crawler searches recent entries using public RSS search engines. From the search result, blogs are discovered as the initial set of seeds. The crawler retrieves RSS files of the seed blogs and pages referred to by the RSS files.

Next for blog discovery, when a crawled page has an HTML link tag that refers to an RSS (this common feature of recent blog tools is called "RSS auto discovery"), the crawler checks whether this RSS represents a blog. If an RSS file satisfies the following conditions, the web site referred to by the RSS as "channel" is recognized as a blog if: (1) The RSS contains items referring to pages in the same host and (2) Each page referred to by the RSS has an HTML link tag that refers back to the RSS.

In entry extraction, the crawler needs to extract entry data from web pages since RSS file does not always contain entire entry content. The crawler extracts the content of an entry from the corresponding web page using an extraction pattern described with XPath expressions. Although the current version requires manual generation and registration of patterns, we plan to develop automated pattern generator similar to the system in [8]. When no extraction pattern is available, the crawler looks for a content snippet in the RSS file.

Subsequently, seed expansion is incorporated to capture possibly-relevant contents. From entry pages of the seed blogs, the crawler crawls hyperlinks for N hops (currently N is set to 1), and discovers blogs. From collected blogs, ones referred to by multiple blogs in the set are chosen for new seeds.

In keyword retrieval, a set of relevant entries are retrieved from the crawled entry data based on keyword matching. Then entries referred to by any entry in the result set are also retrieved and added to the result set. The blog ranking and the tomographic clustering are done on this result set as will be presented in the following section.

5. PRELIMINARY RESULTS

In order to discover and understand the multiple communities in our dataset, we developed the retrieval and exploration system described in Section 3.1. It allows the user to submit a topic query and generates the corresponding blog social network and the mountain view. We begin with subsection 5.1 where the blog ranking results are derived, followed by subsection 5.2 where the discovered communities can be inspected by the mountain view.

5.1 Blog Ranking

For a query, users are interested in finding the top blogs that cover this topic. We build our blog search interface to allow users to input their queries and retrieve the output ranking results. For the input query, users can enter (1) keywords, (2) the period of retrieval, and (3) a selection of either entry or blog ranking outputs. Figure 5 illustrates the blog search interface where we chose the keyword "google" and the period spanning from January 1 to February 22, 2005. Subsequently, relevant entries are scored and blogs are ranked as according to Section 3.2.

In the blog search interface, we can select either entry ranking or blog ranking for our output. Do these two options give us similar or significantly different results? In Figure 5, we illustrate the blog social networks derived from both scoring. On the left social network plot, the blogs corresponding to the top ranked entries are shown. On the right social network, the blogs corresponding to the top ranked blog scores are shown. In this "google" example, it is clear that the social networks are quite different in demonstrating community connectivity. Finally by selecting the blog ranking option, the system generates the list of blogs in ranking order with their corresponding IDs and impact scores, as depicted on the bottom view of Figure 5.

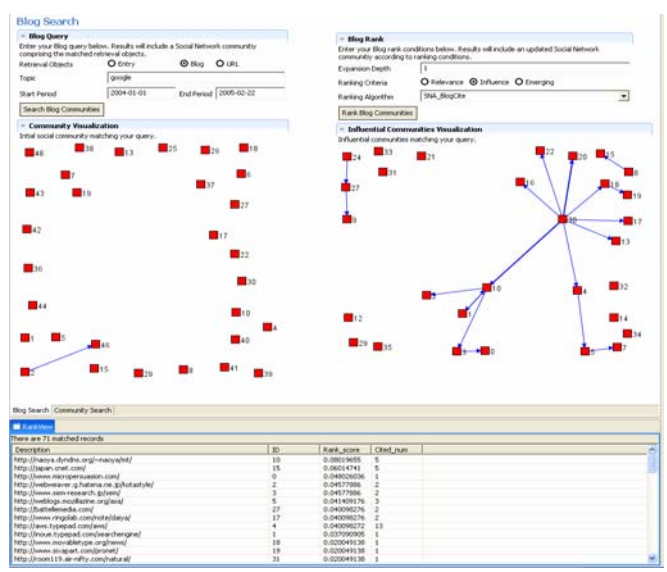

Figure 5. Blog retrieval and ranking for the topic "google". Social network based on entry ranking (left graph) and social network based on blog ranking (right graph).

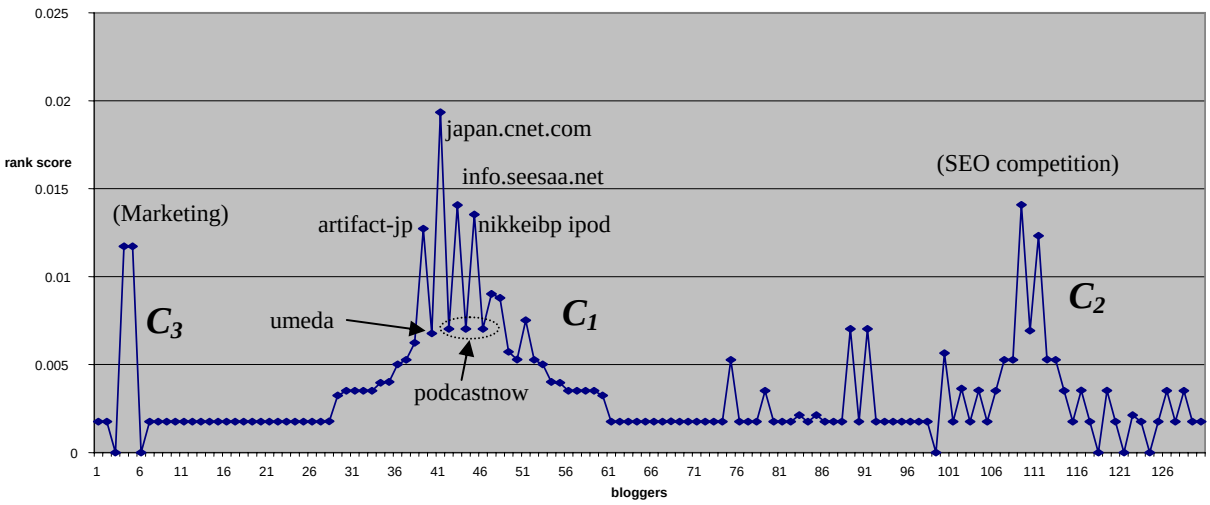

Figure 6. Mountain view of the topic "google". Authorities are shown as peaks, and connectors are shown as valleys.

5.2 Blog Ranking

As demonstrated in the previous subsection, multiple communities are retrieved for a query but it is difficult to discover the significances of these communities other than manually examining each blog. In this section, we will demonstrate how the mountain view can be used to explore some of the communities.

Continuing with our "google" query, our tomographic clustering generates the mountain view depicted in Figure 6. Here we can see three major mountains C_1, C_2, and C_3. In mountain C_1, four outstanding peaks can be seen. Two of those are news sites, *cnet* and *nikkeibp*. One is an announcement blog of a blog hosting site *seesaa* (during this period, *seesaa* announced a new service). One is a personal blog *artifact*, who is famous in the Japanese blog community. These blogs are regarded as authorities and provide informational entries that community members frequently refer to. The blogs, *umeda* and *podcastnow*, connecting these four peaks are regarded as good hubs. They do not only refer to informational authorities but also provide additional values in their entries since their high score shows that their entries are frequently referred to. Mountain C_2 is fairly large but disconnected from the biggest mountain C_1. In fact, it is a community playing a "SEO (Search Engine Optimization) competition." Given an artificial word, they compete for ranking in the Google search result of that word. In order to get high ranking scores, they link to each other creating an intra-connected community but disconnected from outside communities. Mountain C_3 is two blogs on marketing discussing on recent Internet based marketing (such as advertisement in Google).

6. CONCLUSION

Blogs provide an opportunity for people to share important information in a community. In our paper, we provide a mountain view visualization for users to explore the different communities of interest in blogspace. The mountain views are generated using a novel tomographic clustering algorithm based on ranking-based connectivity of the blog social network. Our clustering combines the blog ranking with the inherent connectivity of blog contents. As a result, we capture the landscape of multiple communities and identify representative authorities and connectors.

7. REFERENCES

[1] E. Adar, L. Zhang, L. Adamic, and R. Lukose. "Implicit Structure and the Dynamics of Blogspace," WWW 2004 Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, New York, May 2004.

[2] G. Flake, S. Lawrence, and C. Giles. "Efficient Identification of Web Communities," in Proc. of KDD 2000, ACM Press, New York, 2000, pp. 150-160.

[3] N. Glance, M. Hurst, and T. Tomokiyo. "BlogPulse: Automated Trend Discovery for Weblogs," WWW 2004 Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, New York, May 2004.

[4] D. Gruhl, R. Guha, D. Liben-Nowell, and A. Tomkins. "Information Diffusion Through Blogspace," WWW 2004, New York, May 2004.

[5] J. Kleinberg. "Authoritative sources in a hyperlinked environment," In Proc. 9th ACM-SIAM Symposium on Discrete Algorithms, 1998.

[6] R. Kumar, J. Novak, P. Raghavan, and A. Tomkins. "On the Bursty Evolution of Blogspace," WWW 2003, Budapest, Hungary, May 2003.

[7] R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins. "Trawling the Web for Emerging Cyber-Communities," WWW 1999, Toronto, Canada, May 1999.

[8] T. Nanno, Y. Suzuki, T. Fujiki, and M. Okumura. "Automatic Collection and Monitoring of Japanese Weblogs," WWW 2004 Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics, New York, May 2004.

[9] L. Page, S. Brin, R. Motwani and T. Winograd. "The PageRank Citation Ranking: Bringing Order to the Web," Stanford Digital Libraries Working Paper, 1998.

[10] M. Toyoda and M. Kitsuregawa. "Creating a Web community chart for navigating related communities," Proceedings of the twelfth ACM conference on Hypertext and Hypermedia, Denmark, August 2001.