Richard Freeman, "Topological Tree Clustering of Social Network Search Results" in Proceedings of the Eight International Conference on Intelligent Data Engineering and Automated Learning (IDEAL'07), Lecture Notes in Computer Science (LNCS 4481), Springer, 16-19 December, 2007, pp. 760-769

Topological Tree Clustering of Social Network Search Results

Richard Freeman

Capgemini, Business Information Management
richard.freeman@capgemini.com
http://www.rfreeman.net

Abstract. In the information age, online collaboration and social networks are of increasing importance and quickly becoming an integral part of our lifestyle. In business, social networking can be a powerful tool to expand a customer network to which a company can sell products and services, or find new partners / employees in a more trustworthy and targeted manner. Identifying new friends or partners, on social networking websites, is usually done via a keyword search, browsing a directory of topics (e.g. interests, geography, or employer) or a chain of social ties (e.g. links to other friends on a user's profile). However there are limitations to these three approaches. Keyword search typically produces a list of ranked results, where traversing pages of ranked results can be tedious and time consuming to explore. A directory of groups / networks is generally created manually, requires significant ongoing maintenance and cannot keep up with rapid changes. Social chains require the initial users to specify metadata in their profile settings and again may no be up to date. In this paper we propose to use the topological tree method to dynamically identify similar groups based on metadata and content. The topological tree method is used to automatically organise social networking groups. The retrieved results, organised using an online version of the topological tree method, are discussed against to the returned results of a social network search. A discussion is made on the criterions of representing social relationships, and the advantages of presenting underlying topics and providing a clear view of the connections between topics. The topological tree has been found to be a superior representation and well suited for organising social networking content.

Keywords: Information retrieval, social networking website, social networks, Web 2.0, semantic web, search engine optimization, document clustering, self organizing maps, topological tree, neural networks, post retrieval clustering, taxonomy generation, enterprise content management, enterprise search, information management.

1 Introduction

The exploding growth of web content is leading to an information overload, in which the use of web search engines is becoming critical to finding and retrieving relevant content. Despite the numerous advances in information visualisation [1], the most popular way of presenting search results still remain ranked lists. In this format, the user generally never looks beyond the first three pages, after which they will rather refine their search query by adding more terms or refining the initial query [2]. On the Web the results returned from web search engines, have been widely studied and Search Engine Optimisation (redesigning a website to improve its web pages ranking)

is still a thriving industry [3]. However on social network websites less investigation has been made due to the complexity of the social ties and groups.

Recent research has been focusing on searching social network websites [4], as these have emerged as hugely popular and fast attracting a growing number of users. Some of the site are specialised on different areas, e.g. business (LinkedIn[1]), school/work based (FaceBook[2]), photo sharing (Flickr[3]) or general topics (MySpace[4]). Users generally upload content (e.g. photographs, videos, documents), post comments (blogs, discussions, bookmarks) and biographical information (e.g. name, university attended, current employer), and can often network as friends with other users of groups. With the rapidly growing number of profiles, groups and links, searches on social networking websites are emerging as an important tool for users to for example to find friends, business partners or groups with similar interests. Such searches currently heavily rely on metadata assigned by the user, e.g. a user in Facebook[5] has to provide biographical and personal details for other users to be able to retrieve their profile. One common approach to categorising the user profiles has been to allow them to join communities. This is an important aspect of social networks as these communities help bind the users together. These communities, networks or groups can be based on common interests, activities, or current school/work location.

There are generally three types of ways for a new user to join a community. The first is via browsing a directory where the communities have been manually organised by topics. However this is generally done manually leading to issues around maintenance and subjective interpretation. The second is done by a set of social ties or via recommendations, but such links have to be created manually. The third can be performed via a keyword and/or metadata search. Although ranking mechanisms help order the group profiles in terms of their relevance to the users query, they do not provide any guide as to the overall themes described in the pages or their relationships. Combining the approaches of search and directories has been done for some time in web search engines. For example some efforts have been made to provide different visual representation of the search results, such as suggesting keywords to refine the search (e.g. Webcrawler[6]), representing a graph view of the relations between pages (e.g. Kartoo[7]) or clustering the results (Vivisimo[8]). A major review of the methods and algorithms can be found in [5][6].

This paper deals with methods that organise groups (retrieved by a social network search engine) into a set of virtual folders which are labelled automatically using extracted keywords. A method which clusters group pages dynamically, whilst creating a topology between them in a tree view, is presented in this paper. The topological tree method, first introduced by the author [5], is enhanced through weighting terms depending on their relation to the query term and making the algorithm function efficiently with dynamic social network datasets. Results and discussions confirm that the topological tree representation can be used to provide a

[1] http://www.linkedin.com/
[2] http://www.facebook.com/
[3] http://www.flickr.com/
[4] http://www.myspace.com/
[5] http://www.facebook.com/
[6] http://www.webcrawler.com/
[7] http://www.kartoo.com/
[8] http://www.vivisimo.com/

user with a more intuitive and natural representation for browsing groups and discovering their underlying topics.

2 Visual Representation of Retrieved Social Content

One of the major growth and success factors of social network websites is that the users are repeatability returning to same website like in a real world community. One of the key factors for this being that the content is regularly updated. The content is a key driver that can be accessed through the networks, groups and user profiles. The access to such social content is typically performed using a search query, browsing through a directory, of though the social ties as shown in Table 1.

Table 1. Comparison of the differing methods for accessing the social content.

Name	Description	Issues
Keyword / metadata search	Searches can be made on the name but also the content labels (e.g. photographs, videos, documents), posted comments (blogs, discussions, bookmarks), biographical information (e.g. interests, hobbies, university attended, current employer, geography) and groups (e.g. the groups the users decided to join).	The search is heavily reliant on manual tagging / labelling and the search results are usually presented in a ranked list.
Browsing the group, network or community directory	Typical social networking systems allow a user to visualise and browse a network directory of potential new friendships based on shared interests (e.g. creating new business contacts or finding romantic relationships).	The taxonomy directory might not be adequate for categorising all groups / networks, has to be manually maintained and can rapidly become out of date.
Chain of social ties	A chain of social ties can be used to link friends of friends, e.g. a user can discover friends in common through looking at such social ties. Previous work has suggested that social ties which link two individual are between five and seven [9].	The links are added manually and the search process can be time consuming and might have to be repeated if new links are formed or new profiles are added.

2.1 Searching for profiles and groups in a social network

A number of papers have been published on searching around the Semantic Web which has strong ties with social network searching. For example one author suggested using semantic web analysis in the analysis of social networking website [10]. Community analysis helps finds community structure in social networks. For example the CNM algorithm, is a bottom-up greedy agglomerative clustering which selects and merges pairs of clusters by trying to maximise modularity of the community structure [11]. A more scalable community analysis algorithm has also been proposed [12].

Communities in Orkut (now owned by Google) have previously been analysed through the use of different similarity measure [13]. The main objectives were to evaluate different community similarity measure and recommended a ranked list of related communities, relative to a base community, that might be of interest to some users. The paper found that *L-2 Norm* (also known as cosine distance) showed the best empirical results. Another important finding was the impact of community size on the similarity measures, e.g. Mutual Information favours very large communities, while *L-1 Norm* favours small communities.

2.2 The Importance of Clustering and Topology

In information access systems, the major visual representations are Self-Organising Maps (SOMs), binary trees, balanced or unbalanced trees, graphs, and ranked lists. In some cases a combination of these representations can be used. This section describes the limitations of these methods, and illustrates the benefits of using the topological tree structure.

Clustering algorithms can be used to sort content into categories which are discovered automatically based on a similarity criterion. Its typical output representation is a binary tree or hierarchy. Binary trees quickly become too deep as each level only has two nodes; this representation has been used for retrieval rather than browsing. Hierarchies are typically generated using divisive partitioning algorithms (e.g. divisive *k*-means), or manually constructed such as with social bookmarks / folksonomies (e.g. Del.icio.us[9]) and web directories (e.g. Dmoz[10]). Web directories are particularly beneficial to users who are not familiar with the topics and their relations. However, even if some show cross links with related topics, they do not show the relations between topics at the same level, rather the topics are sorted alphabetically or by popularity. Other search engines such as Vivisimo do cluster results, however at each level in the tree there is always a category "other topics" where many document are clustered to. In addition, as with the other unbalanced trees, there is no relationship between the topics at each level.

Graph representations or SOMs can be used to compensate for this lack of topology in these tree representations or taxonomies. Graphs can represent hyperlinks, relationships or links between topics. A web example of a graph generated representation is Kartoo. Other knowledge representations such as Topic Maps (e.g.

[9] http://del.icio.us/
[10] http://www.dmoz.org/

Omnigator[11]), can also be represented as graph structures. Although they do capture the inter topic / document relations, the major drawback is that they cannot scale easily, i.e. the more nodes / links are added the less legible it becomes. SOMs typically have a 2-dimensional grid structure which adapts to the content space and the number of nodes need not change to represent the underlying number of topics. The SOM-based methods have two distinct properties over other methods, namely non-linear dimensionality reduction and topology preservation. The non-linear projection property ensures that the input space is mapped onto a lower dimensional space with minimum information distortion. The topology preserving clustering enables documents that are similar to be located closely on the map. However one the major weakness of 2-dimensional SOMs, is it is difficult to navigate between different levels of detail. Hierarchical variants of the SOM, such as the Growing Hierarchical SOM [7] have been developed for this purpose; however only one map can be shown at any time and their size is sensitive to fixed parameters. In addition, tables or complex graphics are required to represent the 2-dimensional maps efficiently.

The topological tree method, first proposed by the author [5], compensates for all these factors by exploiting a simple tree view structure to represent both *hierarchical* and *topological relationships* between topics. Previous work undertaken by the author focused on clustering a fixed set of documents. This paper deals with the clustering of search results of multi author / non-uniform documents with different formatting and content. The topological tree can be used to combine the tree structure typically used in file explorers with that of the topology inherent in SOMs. The tree structure allows a user to visualise different levels of detail and hierarchical relationships. The topology, a novel feature specific to the topological trees and SOMs, additionally allows the viewing of the relationships between the topics. Fig. 1 clearly shows the difference between having a topology and not having one. On the left, the topics appear to be randomly placed, but on the right they seem to naturally flow downward as economics, microeconomics, finance, biology, and anatomy making it more intuitive and natural to the user.

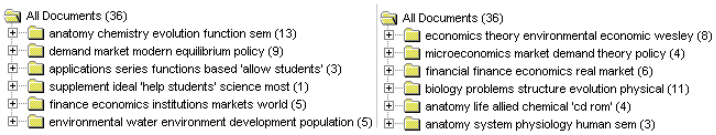

Fig. 1 – *k*-mean with no topology (left) and root level in the topological tree (right)

3 The Topological Tree Method

3.1 Overview of the Method

There are a number of essential steps in the method:

1. The user enters a query term into the local web application, and selects the search options and social network search engine.
2. The application submits the query term to the social search engine and crawls the returned results.

[11] http://www.ontopia.net/omnigator/models/index.jsp

3. Each page is indexed and transformed into a document vector.
4. Feature selection and term weighting is performed on the vector.
5. The documents are organised in a growing chain (see section 3.2).
6. Each chain is labelled and added to the topological tree, if further child chains are required (see section 3.2) return to 4.
7. The user is presented with the resulting generated topological tree.

3.2 Growing Chains and Topological Tree Method

SOMs are generally associated with 2-dimensional structures that help visualise clusters and their relationships in a topology. However, equally 1-dimensional chains can also be used. The topological tree method uses 1-dimensional chains where each node may spawn a child chain. The number of nodes in each chain is guided by an independent validation criterion. The algorithm used to grow the 1-dimensional SOM is termed growing chain (GC) and shares growing properties with the growing grid (used in the GH-SOM [7]) and growing SOM variants, but is more suited for 1-dimension.

In a similar way to the SOM, there are two major steps in the GC algorithm: the search for the best matching unit and the update of the winner and its neighbouring nodes. At time t, an input document vector $\mathbf{x}$ is mapped to a chain consisting of n nodes with a weight vector $\mathbf{w}$. The best matching unit $c(\mathbf{x})$ is the node with the maximum dot product amongst nodes j and document vector $\mathbf{x}(t)$,

$$c(\mathbf{x}) = \arg\max_j \left\{ S_{dot}(\mathbf{x}(t), \mathbf{w}_j) \right\}, \quad j = 1, 2, ...n \tag{1}$$

where n is the current number of nodes. Once the winner node $c(\mathbf{x})$ is found the neighbouring weights are updated using,

$$\mathbf{w}_j(t+1) = \frac{\mathbf{w}_j(t) + \alpha(t) h_{j,c(x)}(t)\mathbf{x}(t)}{\left\| \mathbf{w}_j(t) + \alpha(t) h_{j,c(x)}(t)\mathbf{x}(t) \right\|} \tag{2}$$

where $\alpha(t)$ is the monotonically decreasing learning rate and $h_{j,c(x)}(t)$ the neighbourhood function, typically a Gaussian kernel. When the learning has stabilised for the current number of nodes n, the entropy of the chain is recorded and a new node is inserted next to the node with the highest number of wins. The weights of the new node are initialised by interpolating or extrapolating existing nodes weight values. New nodes are added until n_{max} nodes are reached which corresponds to the maximum allowable chain size. Finally the validation criterion, the entropy-based Bayesian Information Criterion that penalises complexity, gives the optimum number of nodes per chain as:

$$\tau = \arg\min_n \left\{ \frac{1}{m} \sum_{j=1}^{n} m_j \cdot H(C_j) + \frac{1}{2} n \log m \right\}, n = 2, .., n_{\max} \tag{3}$$

where m is the number of documents, n the current number of nodes in the chain, $H(C_j)$ is the total normalised and weighted sum of entropies for cluster C_j.

Then in the hierarchical expansion process, each node in the chain is tested to see if it will spawn a child chain. This is performed using several tests. The first test counts the number of document clustered to that node to see if it is less than a fixed threshold. The next test analyses the vocabulary present in those documents to determine if there is a sufficient number of terms. The final test uses cluster tendency

method. It aims to test if a set of documents contains random documents with no or few relations or if there are strong underlying clusters [8]. If any of these tests fail for a particular node, then it does not spawn a child chain and becomes a leaf node in the final topological tree representation.

Finally each node in the chain is labelled using the most representative terms of the node's weight and its frequency. Once the chain is labelled, then it is added to the current topological tree structure. If further hierarchical expansions in its child chains are required, then the process is repeated for each of the child chains, otherwise the process is terminated and the results presented to the user. The full pre-processing and topological tree method is shown in Fig. 2.

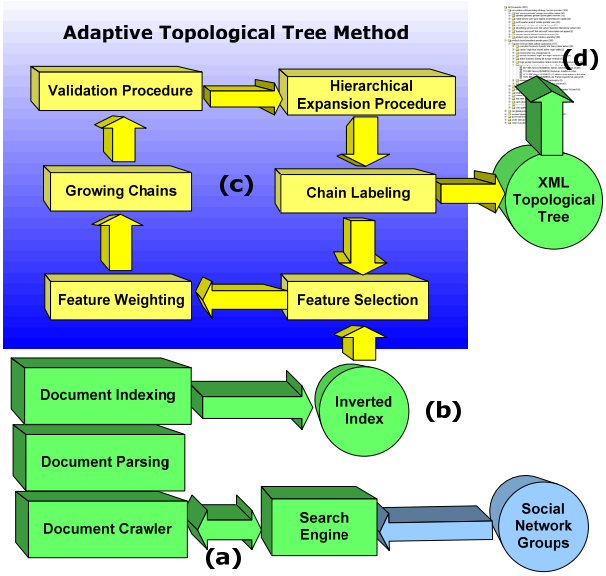

Fig. 2. **The Topological Tree Method.** (a) The search engine is queried, the pages are returned and crawled by the Web Application. (b) An inverted index is generated from the retrieved documents. (c) The closed loop represents the necessary processing for each growing chain in the topological tree. It is grown using an independent validation procedure that estimates the optimum number of nodes that maximise the information value. (d) Once the topological tree is complete it is exported to XML.

4 Results and Discussions

The dataset was dynamically generated from a search query for group pages in MySpace. The query was "developer"; other queries were also tested but omitted for space considerations. The MySpace tree, shown in
Fig. 3, was generated by directly submitting the same query to the search engine and taking a sample snapshot of the results. Fig. 4. shows the topological tree that was generated from running a query and crawling the returned ranked listing.

4.1 Comparison and Discussion

MySpace currently uses simple search criterion of keyword, category and country for retrieving groups. The main limitations of these are that this relies on correct tagging and updating of categories listing. The results can be sorted by newest, most popular (i.e. size) or group name (alphabetical), however a user will still have to browse through pages of results to identify different types of groups they might be interested in. In comparison, the topological tree representation appears more intuitive and natural to the user, as closely related topics are located close to one another in each chain. Each chain does not grow to a large number of nodes, as this number is guided by an independent validation criterion that penalises complexity. In addition hierarchical relations between a parent node and child chain help abstract different levels of detail.

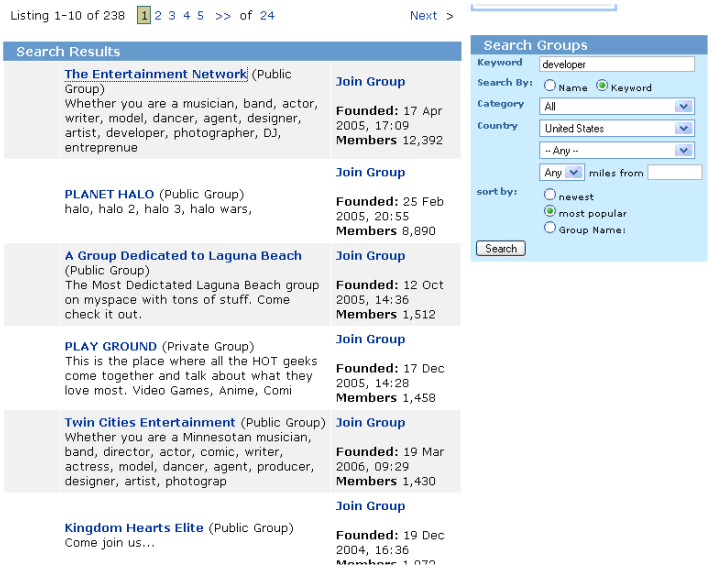

Fig. 3 – A partial snapshot of a search results for groups returned on the query "developer" in MySpace Social networking website. Clearly the sorting by newest, most popular (i.e. size) or group name (alphabetical) is not the best way of organising groups. The user cannot gain an insight of the types of groups without browsing though several pages of the search results or identify the relationship between the groups.

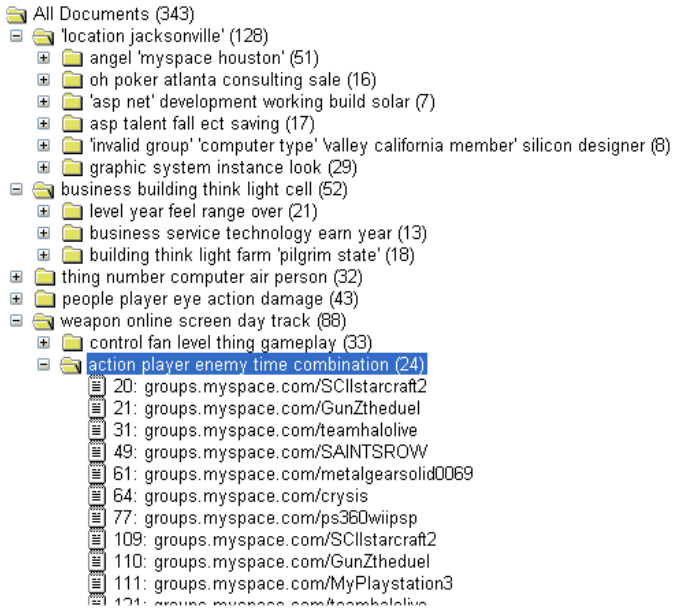

Fig. 4 – A topological tree generated from the pages retrieved using the query "developer" (for all countries). Clearly the topology ensures that the property developer, software developers and games developers are closely grouped together in a more intuitive way.

There are four important criteria for creating an effective browsing experience of documents and topics:

1. *Hierarchical Representation*: the topics need to show different levels of detail simultaneously. This is especially true when the number of topics is large, e.g. the Dewey decimal classification or web directories.
2. *Scalability*: the ability to view a large number of topics and documents in the same window.
3. *Visualise key topics and their related documents*: key topics should be easily be discernable using a label and documents should be shown to belong to one of more of them.
4. *Visualise key relationships*: the ability to visualise the relationships between different topics as well as the connections between documents.

5 Conclusion and Future Work

A topological tree is a tree view structure that does not require complex 2-dimensional graphics or tables such as used in SOM or graphs. Yet it can show the key relationships between extracted topics thus helping reveal previously unknown associations automatically. It also helps make a tree structure appear more intuitive, i.e. related topics are located close to one another in the tree. This topology can be thought of as a graph representation that has been optimised into a tree view, where only the strongest relationships between topics are preserved. Through building on

top of existing search engines, the topological tree method benefits from pre-filtered content where it only needs to organise a relevant subset of the content. This paper has shown that the topological tree can be built on top of a typical social search engine and produce an insightful overview of the underlying topics contained in the top ranking MySpace groups. Future work could look at extracting and combining knowledge from web directories and social networks, with results returned from a web search engine, into a topological tree.

References

[1] Herman, I., Melancon, G., and Marshall, M.: Graph visualization and navigation in information visualization: A survey. Visualization and Computer Graphics, IEEE Transactions on. **6**(1) (2000) 24-43

[2] Search Engine User Behavior Study, White Paper, iProspect, (2006)

[3] Zhang, J., Dimitroff, A.: The impact of webpage content characteristics on webpage visibility in search engine results (Part I), Information Processing and Management **41** (2005) 665–690

[4] Adamic, L., and Adar, E.: How to search a social network, Social Networks **27**(3) (2005) 187-203

[5] Freeman, R.T. and Yin, H.: Adaptive topological tree structure for document organisation and visualisation. Neural Networks. **17**(8-9) (2004) 1255-1271

[6] Freeman, R.T.: Web Document Search, Organisation and Exploration Using Self-Organising Neural Networks, *PhD Thesis*, Faculty of Engineering and Physical Sciences, School of Electrical & Electronic Engineering, University of Manchester: Manchester, (2004)

[7] Rauber, A., Merkl, D., and Dittenbach, M.: The growing hierarchical self-organizing map: exploratory analysis of high-dimensional data. IEEE Transactions on Neural Networks. **13**(6) (2002) 1331-1341

[8] Freeman, R.T. and Yin, H.: Web content management by self-organization. Neural Networks, IEEE Transactions on. **16**(5) (2005) 1256-1268

[9] Dodds, P.S.,: Muhamad, R., Watts D.J., An Experimental Study of Search in Global Social Networks. www.sciencemag.org Science. **301** (2003)

[10] Mika, P.: Flink: Semantic Web Technology for the Extraction and Analysis of Social Networks. **3**(2-3) (2005) 211-223

[11] Clauset, A., Newman, M. E. J., and Moore, C.: Finding community structure in very large networks. *Physical Review E*70:066111 (2004)

[12] Wakita, K., and Tsurumi, T.: Finding Community Structure in Mega-scale Social Networks, WWW 2007, May 8–12, (2007)

[13] Spertus, E., Sahami, M., Buyukkokten, O.: Evaluating Similarity Measures: A Large Scale Study in the Orkut Social Network, KDD '05, August 21-24, (2005) 678-684