SimSpectrum: A Similarity Based Spectral Clustering Approach to Generate a Tag Cloud

Frederico Durao, Peter Dolog, Martin Leginus, and Ricardo Lage

IWIS — Intelligent Web and Information Systems,
Aalborg University, Computer Science Department
Selma Lagerlöfs Vej 300, DK-9220 Aalborg-East, Denmark
{fred,dolog,mlegin09,ricardol}@cs.aau.dk

Abstract. Tag clouds are means for navigation and exploration of information resources on the web provided by social Web sites. The most used approach to generate a tag cloud so far is based on popularity of tags among users who annotate by those tags. This approach however has several limitations, such as suppressing number of tags which are not used often but could lead to interesting resources as well as tags which have been suppressed due to the default number of tags to present in the tag cloud. In this paper we propose the *SimSpectrum:* a similarity based spectral clustering approach to generate a tag cloud which improves the current state of the art with respect to these limitations. Our approach is based on finding to which extent the tags are related by a similarity calculus. Based on the results from similarity calculation, the spectral clustering algorithm finds the clusters of tags which are strongly related and are loosely related to the other tags. By doing so, we can cover part of the tags which are discarded by traditional tag cloud generation approaches and therefore, present the user with more opportunities to find related interesting web resources. We also show that in terms of the metrics that capture the structural properties of a tag cloud such as coverage and relevance our method has significant results compared to the baseline tag cloud that relies on tag popularity. In terms of the overlap measure, our method shows improvements against the baseline approach. The proposed approach is evaluated using MedWorm medical article collection.

Keywords: tag, cloud, medical, information, retrieval, navigation.

1 Introduction

Tag clouds have been popularized as a means for navigation and exploration by social sites, such as *Flickr*, *Technorati* and *del.icio.us*. These sites are used by users to annotate shared resources using short textual labels, called tags. In general, tag annotation are used remembering which meaning the annotated resource had for particular readers or users of the resource and this in a collaborative manner. Aggregated set of the tags form tag clouds. Tag clouds allow users searching for certain tags and to locate resources tagged by these tags also

A. Harth and N. Koch (Eds.): ICWE 2011 Workshops, LNCS 7059, pp. 145–154, 2011.

by other users [10,15]. Tags in the cloud are hyperlinks which users can click and by following the links to see related content. The tags in the tag clouds are mostly presented alphabetically and according to their popularity, i.e. the more a tag was used in annotations of information resources on the Web, the larger the font size it has in a tag cloud. Further, the number of tags in a tag cloud is usually restricted by predefined number which results in cutting out number of tags after the number was reached following the alphabetic order.

We share the same opinion with [18] that "popularity" does not provide the most meaningful groupings to help a user to locate items of interest. For example, if we select the 20 most popular tags assigned to the results of a query "swine flu" at the *MedWorm* portal [1], there might be some articles in the query results that have not been tagged with any of the selected 20 tags and hence the article would not be reachable by the user. The issue is also that despite the popularity, the tags are not necessarily related.

Due to these limitations, we focus on the generation of tag clouds by considering the relatedness of tags. The intention is to partition all tags into disjoint groups of related tags. For instance, the tags "swine", "flu", "mexico", "2010" should be part of one sub cloud while the tags "tumor", "cancer", "blood", "biopsy" should be part of another sub cloud. The sub clouds are treated in this paper as clusters. Our hypothesis is that the organization of the entire tag cloud considering the existence of those sub clouds can better cover and represent the information it links to. Note also, that we are looking for a specific solution to group tags but in this paper, we are not studying how to effectively present those groups for which there are several options. We select only one of the possible presentations for now. The chosen presentation is close to the traditional presentation of tag clouds and only for illustration purposes. The contributions of this paper can be summarized as follows:

- We propose a method which combines *a similarity calculus with a spectral clustering algorithm to generate a tag cloud for navigation and exploration purposes*. We argue for this solution because spectral clustering performs the best in situations where computed clusters should contain strongly related members insight and are very loosely related to the members of other clusters.
- We show that the proposed approach has *promising results in terms of coverage, relevance, and overlap* especially in the context of the very sparse and low quality tagging data set such as that from a medical domain from the *MedWorm portal*. We look especially at this domain as the tag clouds can support surveillance and analysis of information relevant to some medical events such as a disease outbreak. Here the navigation and exploration aids are even more important than in general purpose tagging systems such as *del.icio.us*.

The remainder of this paper is organized as follows. In the next section we review related work on tag cloud systems. Section 3 describes our approach for

[1] http://www.medworm.com/rss/blogtags.php

generation of the tag cloud. Next, Section 4 describes the evaluation, based on the MedWorm dataset. Finally, we conclude the work and point out future works.

2 Related Work

Research on tag clouds has mostly focused on presentation and layout aspects [2,15]. For selecting tags to be displayed in a tag cloud, social information sharing sites mostly use popularity-based schemes. Recently, tag selection algorithms that try to find good and not necessarily popular tags have been developed for structured data [14]. This work relates to our approach in the sense that tag relatedness was also addressed however with less focus on the generation of the tag cloud. There has been extensive research on clustering search results [9,12]. Although not dealing with tag clouds, our work converge to those approaches since all rely on clustering techniques based on tag relatedness. There is also work on query results labeling [13] and categorizing results of SQL queries [5]. [6] adapt tag clouds to provide visual summaries of researchers' activities and use these to promote awareness within a research group. [11] show how tag clouds can be used alongside more traditional query languages and data visualization techniques as a means for browsing and querying databases by both experts and non-expert users. In the same line, Sinclair et al. [16] studied the usefulness of tag clouds versus search interfaces for different types of tasks (general versus specific searches). Similarly to our work, [16] investigate the idea that tag clouds can provide a helpful visual summary of the contents regardless tag popularity. [3] applies tag clustering to overcome the problem of limited search in tag spaces. The difference from our work, is that while we apply the traditional spectral algorithm [19], they combine the spectral bisection algorithm [17] and a modularity function Q, which measures the quality of a particular clustering of nodes in a graph. Technically, the spectral bisection algorithm is a extension of the spectral algorithm that bisects graphs into two graphs. Division into a larger number of graphs is usually achieved by repeated bisection. Another difference in comparison to our work is that we extend the weights for tag relatedness with similarity calculation while [3] only considers co-occurrence of tags. The final difference is that we have also performed an evaluation study based on compactness metrics.

In a medical domain, [8] propose a lightweight technique that uses multiple synchronized tag clouds to support iterative visual analysis and filtering of query results. The proposal was evaluated in a user study which presents typical search and comparison scenarios to users trying to understand heterogeneous clinical trials from a leading repository of scientific information. Unlike our work, they did not use any specific technique for analyzing the relatedness of tags. Therefore, our work provide a better solution for their problem as well. [1] introduce a new model for collaborative tagging in medical blogs, i.e. tagging blog entries with medical information. MTag includes two modules: the service module and the semantic module. The service module enables health professionals provide blog posts with auto-completed tags that represent actual medical terms and categorize their tags. Tags are mapped to URIs from online medical knowledge

datasets to clarify their medical meaning. [4] describe a prototype which retrieves biomedical information from different sources, manages it to improve the results obtained and to reduce response time and, finally, integrates it so that it is useful for the clinician, providing all the information available about the patient at the POC. Moreover, it also uses tools which allow medical staff to communicate and share knowledge.

3 Tag Cloud Approach Based on Spectral Clustering

Figure 1 shows an excerpt of the MedWorm tag cloud (on the left side) and our generated tag cloud (on the right side). The first visible observation is that our tag cloud reduces the amount of tags in the cloud. Tags as "award, awards, Australia, ethics, advocacy" are not considered by our approach since they are not closely related to the other tags in the cloud. The second observation refers to the organization of the cloud itself. In the MedWorm cloud, many unrelated tags are located next to each other. Examples include "aids and alcohol", "awards and back pain" and "advocacy and affairs". In our tag cloud, we organize the "sub clouds" (clusters) per line and provide an allocation of tags based on their relatedness. Examples include "protein and aids" and "alcoholic and addiction". These sets are not found in the MedWorm tag cloud.

Fig. 1. An example of tag clouds from MedWorm dataset, on the left the original popularity based tag cloud and on the right generated by our approach

The tag cloud approach used in the Figure 1 is described bellow. It is based on two main steps: first, it calculates a similarity measure among tags, and, second, it runs a clustering technique on the tag cloud space to identify the sub clouds.

3.1 Calculating Tag Relatedness

We represent the tag space as a similarity matrix W that captures the relatedness of all tags. Since our goal is to find strongly related tags, we use the frequency counts of all the co-occurred tag pairs (co-tags) and attempt to identify the significant co-tags. In order to do that, we determine the pairs of tags that co-occur more frequently, i.e., the pair of tags that are frequently assigned to the same article. In short, W is calculated as:

$$W = \sum |tag_i \cap tag_j|, \tag{1}$$

where $tag_i \in T$ and $tag_j \in T$ and T is the set of tags. The second step is to look for a cutoff point above which the co-tags are considered strongly related. The weakly related co-tags are discarded and not considered in further computations. The cutoff point is calculated based on the analysis of co-tags statistics and it is important to discard the noisy and weakly related co-tags which cause inaccurate clustering.

Once the strongly related co-tags are identified, we compute affinity among co-tags according to a similarity function. Different similarity measures can be exploited, but for this work, we opted for using cosine similarity because we obtained the best results in our preliminaries analysis [7]. The cosine similarity is calculated as follows:

$$Cosine(tag_i, tag_j) = \frac{2|I \cap J|}{\sqrt[2]{|I| \times |J|}} \tag{2}$$

, where the amount of tag occurrences for tag_i and tag_j within all tag assignments is denoted by $|I|$ and $|J|$ respectively. The number of co-occurrences between tag_i and tag_j is given by $|I \cap J|$. This similarity measure is computed for every strongly related co-tag in the tag space, once we can transform the tag pair relations into a graph structure. It is an undirected weighted graph $G(V, E, W)$ consisting of:

- a set of nodes V, where a vertex v_i of the graph corresponds to a tag tag_i.
- a set of edges E, where an edge e_i connects vertices v_i and v_j if the tag tag_i relates strongly to tag tag_j or vice versa.
- weights are given by the affinity matrix W, where a weight $w_{i,j}$ corresponds to the similarity between tag_i and tag_j.

As the graph G is undirected, it holds that $w_{i,j} = w_{j,i}$ and the affinity matrix W is symmetric. The next step is to group similar tags into clusters.

3.2 Clustering Tag Space

Once the graph G is created, we then proceed to find (sub) clusters of tags that address the same topic. For instance, a cluster of tags addressing the topic "diet" could contain the tags "meal", "vitamin", "periodicity", while a cluster of tags addressing the topic "infectious diseases" could contain the tags "contamination", "virus", "oral contact". This requirement matches exactly the principle of spectral clustering algorithms, i.e. to cut a weighted graph into a number of disjoint pieces (clusters) such that the intra-cluster weights (similarities) are high and the inter-cluster weights are low [19]. To obtain clusters, we therefore rely on a spectral clustering algorithm which input is the undirected weighted graph G. The spectral clustering algorithm partitions the graph G based on its spectral decomposition into subgraphs. The affinity matrix W expresses the graph G, in such way that for each node the matrix W contains a row with graph weights (similarities values) between a given node and all other nodes. The steps to run the spectral clustering are:

1. We build the Laplacian matrix $L = D^{-1/2}WD^{-1/2}$ derived from the affinity matrix W. The D is $n \times n$ diagonal matrix whose $(i,i) - th$ element is the sum of W's $i - th$ row, in other words it is degree of a given node i - sum of all weights corresponding to the edges that are connected to a given node i. The Laplacian matrix L is symmetric and has identical size as affinity matrix W.
2. We compute the k largest eigenvectors of L, these obtained top k eigenvectors are used as columns to create a new matrix $U \in \mathcal{R}^{n \times k}$. We consider each row of U as a point in $\mathcal{R}^k$, hence we can apply standard K-means algorithm to cluster these points into k clusters. In our experiment, we empirically tried different numbers of clusters to run our analysis and concluded that for our experiment and the dataset 10 clusters perform the best. This could however differ from a dataset to dataset and can even change with the evolution of the tag set. Thus, an approach that automatically defines the member of clusters is envisaged as part of our future works.
3. Finally, we map original node i to the cluster j if and only if row i from matrix U belongs to the same cluster j. We obtained disjoint groups of similar and related tags and we are able to build enriched tag cloud.

4 Evaluation

4.1 Dataset and Experimental Setup

Methodology. In order to evaluate the generated tag cloud, we analyzed the problem from a traditional information retrieval perspective. We used tags from each sub cloud as query terms and analyzed the search results issued by these tag queries. Indeed, we compared the set of tags assigned to returned results against the set of tags in the each sub cloud (or cluster). In this sense we could calculate three structural properties of the cloud: *coverage*, *overlap* and *relevance*. For issuing the queries, we utilized the *Apache Lucene* [2] as our search engine.

For the matter of comparison, we repeated the same procedure on the MedWorm tag cloud. Since MedWorm's cloud relies on tag popularity and does not deal with explicit clusters, we decided to create "fake" clusters composed by tag neighbors located after and before a tag query q present in our sub clouds. Thus, the clusters were made up around all tag queries common in both clouds. In this sense, we could build clusters of $T_{neighbors}$ for the MedWorm tag cloud and compare the results against our approach. The amount of clusters and size was the same as used in our approach. Regarding the amount of clusters for the cloud, we empirically set the number of cluster based on our observations of the cluster quality. After testing a tag cloud containing 5 to 20 clusters, we ran our experiments with 5, 10 and 15 clusters.

Data and Queries. We crawled medical articles from *MedWorm* repository and stemming out the entity attributes from the data. Thus, we obtained the

[2] http://lucene.apache.org/

tags, resources and its associations. The resulting dataset comprises 13,509 tags and 26,1501 documents. We also indexed the stemmed words from documents to build up the search space. Finally, the tag cloud was pre-processed according to the steps described in Section 3.

As noted before, all tags from the clusters of our generated tag cloud were utilized as individual queries as long as they were also found in the baseline tag cloud. In this sense, both approaches could be evaluated on the same search results. We justify the utilization of tags as queries to avoid using "arbitrary" terms (even medical related), that eventually could not retrieve results and thus not contributing the evaluation.

Evaluation Metrics. The quality of tag cloud has been studied in many studies [10,18]. In this work, we evaluate the quality of our cloud inspired by metrics established by [18]. In particular, we pay special attention to the *coverage*, *overlap* and *relevance* of the cloud. We understand that these metrics capture the structural properties of a cloud and indicate the its quality for representing the collection of tagged documents. In order to formally describe the metrics, let T_c be the set of tags in a cluster c; and C_q be the set of items retrieved when a query q is issued.

- *Coverage of* T_c: Some items in C_q may not be assigned with any tag from T_c. Then, these objects are not covered by T_c. Coverage gives us the fraction of C_q covered by T_c. Thus, coverage $cov(T_c)$ is defined as:

$$cov(T_c) = \frac{|T_c|}{|C_q|}, \tag{3}$$

  This metric can take values between 0 and 1. If $cov(T_c)$ is close to 0, then T_c is associated with a few items of C_q.
- *Overlap of* T_c: Different tags in T_c may be assigned with the same item in C_q. The overlap metric captures the extent of such redundancy. Thus, given $t_i \in T_c$ and $t_j \in T_c$, we define the overlap $over(T_c)$ of T_c as:

$$over(T_c) = avg_{t_i \neq t_j} \frac{|t_i \cap t_j|}{|C_q|}, \tag{4}$$

  This metric also lies in [0,1]. If $over(T_c)$ is close to 0, then the intersections of tags in the same cluster are small and redundancy is minor.
- *Relevance of* T_c: It says how relevant the tags in T_c are to the original query q. To answer this, we treat each t in $T_c - q$ as a query and we consider the set C_t of items that this query returns. Since we decided to use one tag in T_c as q, for obvious reasons, we set the constraint: $t \neq q$. The more C_t and C_q overlap, the more related t is to q. If $C_t \subseteq C_q$, then t is practically a sub-category of the original query q. Let us first define the relevance $rel(t, q)$ of a tag t to the original query q as the fraction of results in C_t that also belong to C_q, i.e.:

$$rel(T_c) = avg_{t \in T_c} \frac{|C_t \cup C_q|}{|C_t|}, \tag{5}$$

The $rel(T_c)$ lies in [0,1]. The closer it is to 1, the more relevant is T_c to the query q.

4.2 Evaluation Results

We generated our cloud based on the baseline MedWorm cloud containing 200 tags. This is the approximate amount of tags available on MedWorm web site. After generating our cloud, only 125 tags were considered. The 75 tags missing were discarded by the clustering algorithm. Only 70 tags from our tag cloud were also found in the baseline tag cloud. All those 70 tags were used as queries in the evaluation. Table 1 shows the comparative results of our analysis taking into account the three aforementioned metrics. The results on the left side of the table refer to MedWorm tag cloud while the results on the right side of the table are achieved from our approach. The results correspond to the mean values for the metrics assessed. As results show, our approach obtained significant advantage

Table 1. *Mean Values* for the Metrics Assessed

	MedWorm Tag Cloud			Our Tag Cloud		
# Cluster	Coverage	Relevance	Overlap	Coverage	Relevance	Overlap
5	0.53	0.56	0.65	0.67	0.66	0.61
10	0.51	0.55	0.67	0.70	0.68	0.61
15	0.56	0.52	0.61	0.65	0.63	0.59

(on average) in terms of coverage and relevance at rates of 20.4% and 16.4% respectively. We also achieved better overlap rates than the MedWorm tag cloud at a satisfactory rate of 5.7%. As to the number of clusters, we observed best results with 10 clusters were considered.

Focusing exclusively on the coverage metric, we can argue that reorganization of the cloud in sub clusters of related terms made it more representative. This means that each tag covers a more expressive part of the whole indexed corpus. Although no significant improvements were observed for overlap, at least we could observe that tags assigned to the search results were more equally distributed thus reducing scarcity. The immediate benefit is that searchers if using the tags as queries might increase the chances for hits of desired documents. As to the relevance metric, we can argue that the clusters contribute to generate a more cohesive cloud that better cover and represent the information it links to. We outline two benefits of our approach: i)it demonstrates how closely related the tags are in the cloud and ii) how closely related the search results are to the sub clouds.

5 Conclusion and Future Works

In this paper we propose an approach to generate quality tag clouds by considering the relatedness of tags and separation of concerns. Our hypotheses was that

the organization of the whole cloud considering the relatedness of tags could improve structural properties of the cloud and thereby enhance information retrieval capabilities.

According to our results, we reached higher levels of coverage, overlap and relevance compared to a baseline medical tag cloud. As a future work, we plan to investigate how different metrics may correlate to each other in order to determine which independent metrics make sense as optimizations objectives. In addition, it is possible that metrics may exhibit different correlation trends in different datasets. As previously said, we plan to utilize an clustering approach that automatically define the amount of clusters. Further, digital dictionaries as WordNet or even domain ontologies should be considered for calculation of tag relatedness. We also plan to compare the clustering algorithm using the bisection technique with the one used in this work. Finally, a task-based evaluation using a navigation tool is planned to better support the validity of the approach.

Acknowledgment. This work has been supported by FP7 ICT project M-Eco: Medical Ecosystem Personalized Event-Based Surveillance under grant number 247829 and FP7 ICT project KiWi: Knowledge in a Wiki under grant agreement No. 211932.

References

1. Batch, Y., Yusof, M.M., Noah, S.A.M., Lee, T.P.: Mtag: A model to enable collaborative medical tagging in medical blogs. Procedia Computer Science 3, 785–790 (2011); World Conference on Information Technology
2. Bateman, S., Gutwin, C., Nacenta, M.: Seeing things in the clouds: the effect of visual features on tag cloud selections. In: Proceedings of the Nineteenth ACM Conference on Hypertext and Hypermedia, HT 2008, pp. 193–202. ACM, New York (2008)
3. Begelman, G., Keller, P., Smadja, F.: Automated tag clustering: Improving search and exploration in the tag space. In: Proceedings of the WWW Collaborative Web Tagging Workshop, Edinburgh, Scotland (2006)
4. Cabarcos, A., Sanchez, T., Seoane, J.A., Aguiar-Pulido, V., Freire, A., Dorado, J., Pazos, A.: Retrieval and management of medical information from heterogeneous sources, for its integration in a medical record visualisation tool. IJEH 5(4), 371–385 (2010)
5. Chakrabarti, K., Chaudhuri, S., Hwang, S.-W.: Automatic categorization of query results. In: Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data, SIGMOD 2004, New York, NY, USA, pp. 755–766 (2004)
6. de Spindler, A., Leone, S., Geel, M., Norrie, M.C.: Using Tag Clouds to Promote Community Awareness in Research Environments. In: Luo, Y. (ed.) CDVE 2010. LNCS, vol. 6240, pp. 3–10. Springer, Heidelberg (2010)
7. Durao, F., Lage, R., Dolog, P., Coskun, N.: Exploring multi-factor tagging activity for personalized search. In: WEBIST 2011, Proceedings of the 7th International Conference on Web Information Systems and Technologies, The Netherlands, May 6-9 (2011)

8. Hernandez, M.-E., Falconer, S.M., Storey, M.-A., Carini, S., Sim, I.: Synchronized tag clouds for exploring semi-structured clinical trial data. In: Proceedings of the 2008 Conference of the Center for Advanced Studies on Collaborative Research: Meeting of Minds, CASCON 2008, pp. 4:42–4:56. ACM, New York (2008)
9. Koutrika, G., Zadeh, Z.M., Garcia-Molina, H.: Coursecloud: summarizing and refining keyword searches over structured data. In: Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology, EDBT 2009, pp. 1132–1135. ACM, New York (2009)
10. Kuo, B.Y.-L., Hentrich, T., Good, B.M., Wilkinson, M.D.: Tag clouds for summarizing web search results. In: Proceedings of the 16th International Conference on World Wide Web, WWW 2007, pp. 1203–1204. ACM, New York (2007)
11. Leone, S., Geel, M., Muller, C., Norrie, M.C.: Exploiting tag clouds for database browsing and querying. In: Aalst, W., Mylopoulos, J., Rosemann, M., Shaw, M.J., Szyperski, C., Soffer, P., Proper, E. (eds.) Information Systems Evolution. LNBIP, vol. 72, pp. 15–28. Springer, Heidelberg (2011)
12. Maslowska, I.: Phrase-Based Hierarchical Clustering of Web Search Results. In: Sebastiani, F. (ed.) ECIR 2003. LNCS, vol. 2633, pp. 555–562. Springer, Heidelberg (2003)
13. Nigam, K., McCallum, A.K., Thrun, S., Mitchell, T.: Text classification from labeled and unlabeled documents using em. Mach. Learn. 39, 103–134 (May 2000)
14. Rivadeneira, A.W., Gruen, D.M., Muller, M.J., Millen, D.R.: Getting our head in the clouds: toward evaluation studies of tagclouds. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI 2007, pp. 995–998. ACM (2007)
15. Schrammel, J., Leitner, M., Tscheligi, M.: Semantically structured tag clouds: an empirical evaluation of clustered presentation approaches. In: Proceedings of the 27th International Conference on Human Factors in Computing Systems, CHI 2009, pp. 2037–2040. ACM (2009)
16. Sinclair, J., Cardew-Hall, M.: The folksonomy tag cloud: when is it useful? J. Inf. Sci. 34, 15–29 (2008)
17. Van Driessche, R., Roose, D.: An improved spectral bisection algorithm and its application to dynamic load balancing. Parallel Comput. 21, 29–48 (1995)
18. Venetis, P., Koutrika, G., Garcia-Molina, H.: On the selection of tags for tag clouds. In: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, WSDM 2011, pp. 835–844 (2011)
19. von Luxburg, U.: A tutorial on spectral clustering. Statistics and Computing 17, 395–416 (2007)