FACETED RANKING IN COLLABORATIVE TAGGING SYSTEMS
Efficient Algorithms for Ranking Users based on a Set of Tags

José I. Orlicki(1,2), Pablo I. Fierens(2) and J. Ignacio Alvarez-Hamelin(2,3)
(1) Core Security Technologies, Humboldt 1967 1°p, C1414CTU Buenos Aires, Argentina
(2) ITBA, Av. Madero 399, C1106ACD Buenos Aires, Argentina
(3) CONICET (Argentinian Council of Scientific and Technological Research)

Keywords: Faceted search, User reputation, Collaborative tagging, Social bookmarking, Web graph.

Abstract: Multimedia content is uploaded, tagged and recommended by users of collaborative systems such as *YouTube* and *Flickr*. These systems can be represented as tagged-graphs, where nodes correspond to users and tagged-links to recommendations. In this paper we analyze the online computation of user-rankings associated to a set of tags, called a *facet*. A simple approach to faceted ranking is to apply an algorithm that calculates a measure of node centrality, say, PageRank, to a subgraph associated with the given facet. This solution, however, is not feasible for *online* computation. We propose an alternative solution: (i) first, a ranking for each tag is computed *offline* on the basis of tag-related subgraphs; (ii) then, a faceted order is generated *online* by merging rankings corresponding to all the tags in the facet. Based on empirical observations, we show that step (i) is scalable. We also present efficient algorithms for step (ii), which are evaluated by comparing their results to those produced by the direct calculation of node centrality based on the facet-dependent graph.

1 INTRODUCTION

In collaborative tagging systems, users assign keywords or *tags* to their uploaded content or bookmarks, in order to improve future navigation, filtering or searching (Marlow et al., 2006). These systems generate a categorization of content commonly known as a *folksonomy*.

Two well-known collaborative tagging systems for multimedia content are *YouTube* [1] and *Flickr* [2], which are analyzed in this paper. These systems can be represented as tagged-graphs such as the one shown in Figure 1. In this example, there are four users, *A*, *B*, *C* and *D*. **M** is the set of contents and associated tags. For example, user *B* has uploaded one multimedia content, *song2*, to which it has associated the tag-set {*blues*,*jazz*}. **V** is the set of recommendations; e.g., user *A* recommends *song2* of user *B*, which is represented in the graph as an arrow from *A* to *B* with tags *blues*,*jazz*.

Users can be ranked in relation to a set of tags, which we call a *facet*. Some applications of *faceted* (i.e., tag-associated) rankings are: (i) searching for content through navigation of the best users with respect to a facet; (ii) measuring reputation of users by listing their best rankings for different tags or tag-sets. Both applications lead to detection of field experts and web personalization.

M = {(*A*, *song1*, {*blues*})
(*B*, *song2*, {*blues*,*jazz*})
(*C*, *song3*,{*blues*})
(*C*, *song4*,{*jazz*})
(*D*, *song5*,{*blues*})
(*D*, *song6*,{*rock*}) }

V = {(*A*, *song2*)
(*B*, *song4*)
(*B*, *song5*)
(*A*, *song3*)
(*A*, *song4*)
(*C*, *song6*) }

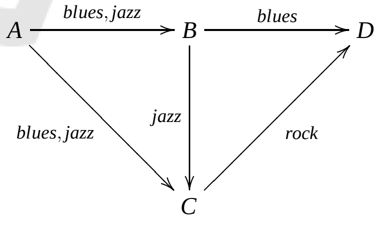

Figure 1: Example of construction of a tagged-graph from a set of contents **M** and a set of recommendations **V**.

The order or ranking can be determined by a centrality measure, such as PageRank (Page et al., 1998;

[1] http://www.youtube.com/

[2] http://www.flickr.com/

I. Orlicki J., I. Fierens P. and Ignacio Alvarez-Hamelin J.
FACETED RANKING IN COLLABORATIVE TAGGING SYSTEMS - Efficient Algorithms for Ranking Users based on a Set of Tags.
DOI: 10.5220/0001826306210628
In *Proceedings of the Fifth International Conference on Web Information Systems and Technologies (WEBIST 2009)*, page
ISBN: 978-989-8111-81-4

Langville and Meyer, 2003), applied to a recommendation graph. Given a facet, a straightforward solution is to compute the centrality measure based on an appropriate facet-dependent subgraph of the recommendation graph. However, *online* computation of the centrality measure is unfeasible because its high time complexity, even for small facets with two or three tags. Moreover, the offline computation of the centrality measure for each facet is also unfeasible because the number of possible facets is exponential in the number of tags. Therefore, alternative solutions must be looked for. A simple solution is to use a general ranking computed offline, which is then filtered online for each facet-related query. The use of a single ranking of web pages or users within folksonomies has the disadvantage that the best ranked ones are those having the highest centrality in a global ranking which is facet-independent. In the information retrieval case, this implies that the returned results are ordered in a way which does not take into account the focus on the searched topic. This problem is called *topic drift* (Richardson and Domingos, 2002).

In this paper we propose a solution to the problem of topic drift in faceted rankings which is based on PageRank as centrality measure. Our approach follows a two-step procedure: (i) a ranking for each tag is computed offline on the basis of a tag-related subgraph; (ii) a faceted order is generated online by merging rankings corresponding to all tags in the facet.

The fundamental assumption is that step (i) in this procedure can be computed with an acceptable overhead which depends on the size of the dataset. This hypothesis is validated by two empirical observations. On one hand, in the studied tagged-graphs of *Flickr* and *YouTube*, most of the tags are associated to very small subgraphs, while only a small number of tags have large associated subgraphs (see Sect. 3). On the other hand, the mean number of tags per edge is finite and small as explained in Sect. 4.1.

The problem then becomes to find a *good* and *efficient* algorithm to merge several rankings in step (ii), and we devote Sect. 4 to that task. The *"goodness"* of a merging algorithm is measured by comparing its results to those produced by the naive approach of applying the PageRank algorithm on facet-dependent graphs (see Sect. 5). The *efficiency* of an algorithm is evaluated by means of its time complexity.

We concentrate our effort on facets that correspond to the *logical conjunction* of tags (*match-all-tags*-queries) because this is the most used logical combination in information retrieval (Christopher, 2008).

The remaining of the paper is organized as follows. We discuss prior works and their limitations in Sect. 2. In Sect. 3 we explore two real examples of tagged-graphs. In particular, we analyze several important characteristics of these graphs, such as the scale-free behavior of the node indegree and assortativeness of the embedded recommendation graph (see Sect. 3.1). Several rank-merging algorithms are introduced in Sect. 4 and their scalability is analyzed in Sect. 4.1. We discuss experimental results in Sect. 5 and we conclude with some final remarks and possible directions of future work in Sect. 6.

2 RELATED WORK

Theory and implementation concepts used in this work for PageRank centrality are based on the comprehensive survey in (Langville and Meyer, 2003). Basic topic-sensitive PageRank analysis was attempted biasing the general PageRank equation to special subsets of web pages in (Al-Saffar and Heileman, 2007), and using a predefined set of categories in (Haveliwala, 2002). Although encouraging results were obtained in both works, they suffer from the limitation of a fixed number of topics biasing the rankings. In other variations of personalized PageRank, the ranking was augmented with weights based on usage (Eirinaki and Vazirgiannis, 2005) and on access time length and frequency by previous users (Guo et al., 2007). However, these approaches built a unique PageRank which was neither user dependent nor query dependent.

(Hotho et al., 2006) adapted PageRank to work on a tripartite graph of users, tags and resources corresponding to a folksonomy. They also developed a form of topic-biasing on the modified PageRank, but the generation of a faceted ranking implied a new computation of the adapted algorithm on the graph for each new facet.

Recent advances on collaborative tagging systems include the extraction of more structured metadata from these systems (Al-Khalifa et al., 2007) and the introduction of groups of tagged resources with their appropriate tagging by users that carries new valuable information about the structure of Web content (Abel et al., 2008).

There is a broad literature on the automatic discovery of topics of interest, e.g., (Li et al., 2008). In this paper, however, we focus on user-selected facets (topics).

There has also been some work done on faceted ranking of web pages. For example, the approach of (DeLong et al., 2006) involves the construction of a larger multigraph using the hyperlink graph with each node corresponding to a pair webpage-concept and each edge to a hyperlink associated with a concept. Although (DeLong et al., 2006) obtain good ranking results for single-keyword facets, they do not support multi-keyword queries.

Query-dependent PageRank calculation was introduced in (Richardson and Domingos, 2002) to extract a weighted probability per keyword for each webpage. These probabilities are summed up to generate a query-dependent result. They also show that this faceted ranking has, for thousands of keywords, computation and storage requirements that are only approximately 100-200 times greater than that of a single query-independent PageRank. As we show in Sect. 4.1, the *offline* phase of our facet-dependent ranking algorithms has similar time complexity.

Scalability issues were tackled by (Jeh and Widom, 2002) criticizing offline computation of multiple PageRank vectors for each possible query and preferring another more efficient dynamic programming algorithm for online calculation of the faceted rankings based on offline computation of basis vectors. They found that their algorithm scales well with the size of the biasing page set.

As in (Jeh and Widom, 2002) we also avoid computing offline the rankings corresponding to all possible facets and our solution requires only the offline computation of a ranking per tag. A faceted ranking is generated online by adequately merging the rankings of the corresponding tags. Sect. 4 deals with different approaches to the merging step.

3 TAGGED-GRAPHS

In this section, we present two examples of collaborative tagging systems, *YouTube* and *Flickr*, where content is tagged and recommendations are made. Although these systems actually rank content, to our knowledge, no use of graph-based faceted ranking is made.

The taxonomy of tagging systems in (Marlow et al., 2006) allows us to classify *YouTube* and *Flickr* in the following ways: regarding the tagging rights, both are *self-tagging* systems; with respect to the aggregation model, they are *set* systems; concerning the object-type, they are called *non-textual* systems; in what respects to the source of material, they are classified as *user-contributed*; finally, regarding tagging support, while *YouTube* can be classified as a *suggested* tagging system, *Flickr* must be considered a *blind* tagging system.

In our first example the content is multimedia in the form of favorite videos recommended by users. We collected information from the service *YouTube* using the public API crawling 185,414 edges and 50,949 nodes in Breadth-First Search (BFS) order starting from the popular user *jcl5m* that had videos among in the top twenty rated ones during April 2008. We only considered nodes with indegree greater than one, because they are the relevant nodes to PageRank. ¿From this information, we constructed a full tagged-graph G. We have also constructed subgraphs by preserving only those edges that contain a given tag (e.g., $G(music)$ and $G(funny)$ corresponding to the tags $music$ and $funny$, respectively), any tag in a set (e.g., $G(music \vee funny)$) or all tags in a set (e.g., $G(music \wedge funny)$). Table 1 presents the number of nodes and edges of each of these graphs. We must note that mandatory categorical tags such as $Entertainment$, $Sports$ or $Music$, always capitalized, were removed in order to include only tags inserted by users.

Table 1: Sizes of *YouTube*'s graph and some of its subgraphs.

Graph	nodes	edges
G	50,949	185,414
$G(music \vee funny)$	4,990	13,662
$G(music)$	2,650	5,046
$G(funny)$	2,803	6,289
$G(music \wedge funny)$	169	188

In our second example the content are photographs and the recommendations are in the form of favorite photos[3]. We collected information from the service *Flickr* by means of the public API, crawling 225,650 edges and 30,974 nodes in BFS order starting from the popular user *junku-newcleus* and including only nodes with indegree grater than one. The full tagged-graph G and the sample subgraphs $G(blue \vee flower)$, $G(blue)$, $G(flower)$ and $G(blue \wedge flower)$ were constructed. The number of nodes and edges of these graphs are shown in Table 2.

Table 2: Sizes of *Flickr*'s graph and some of its subgraphs.

Graph	nodes	edges
G	30,974	225,650
$G(blue \vee flower)$	5,440	14,273
$G(blue)$	3,716	6,816
$G(flower)$	2,771	6,370
$G(blue \wedge flower)$	280	327

[3] Only the first fifty favorites photos of each user were retrieved.

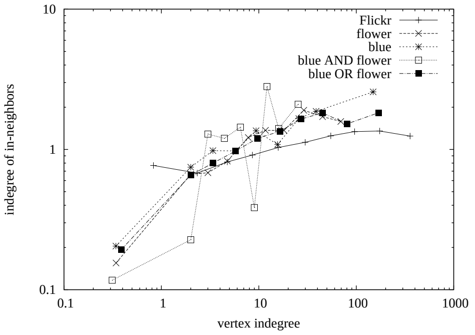

Figure 2: Binned correlation of indegree of in-neighbors with indegree.

3.1 Graph Analysis

We have verified that node indegree, in both *YouTube* and *Flickr* graphs, is characterized by a power-law distribution: $P(k) \approx k^{-\gamma}$, with $2 < \gamma < 3$. Experience with Internet related graphs shows that the power-law distribution of the indegree does not change significantly as the graph grows and, hence, the probability of finding a node with an arbitrary degree eventually becomes non-zero (Pastor-Satorras and Vespignani, 2004).

Since recommendation lists are made by individual users, node outdegree does not show the same kind of scale-free behavior that node indegree. On the contrary, each user recommends only 20 to 30 other users on average. Moreover, since node outdegree is mostly controlled by *human* users, we do not expect its average to change significantly as the graph grows.

In *YouTube*'s graph there is no clear correlation of indegree of in-neighbors with node indegree, but in *Flickr*'s graph there is a slight assortativeness (Newman, 2002), indicating a biased preference of nodes with high indegree for nodes with high indegree (see Figure 2).

4 FACETED RANKING

Given a set **M** of tagged content, a set **V** of favorite recommendations and a tag-set or facet **F**, the *faceted ranking problem* consists on finding the ranking of users according to facet **F**. The naive solution is to find a graph associated to the facet and apply the PageRank algorithm to it. This approach leads to two algorithms, called **edge-intersection** and **node-intersection**, which turn out to too costly for online queries. Indeed, their computation requires the extraction of a subgraph which might be very large in a large graph[4] and the calculation of the corresponding PageRank vector. However, they serve as a basis of comparison for more efficient algorithms.

Edge-intersection. Given a set of tags, a ranking is calculated by computing the centrality measure of the subgraph corresponding to the recommendation edges which include *all* the tags of a certain facet. The main idea is to find those edges having all tags and then compute PageRank only on this subgraph which yields the *relevant nodes*.

Node-intersection. Consider the example given in Fig. 1 under the query $blues \wedge rock$. According to the **edge-intersection** algorithm, there is no node in the graph satisfying the query. However, it is reasonable to return node D as a response to such search. In order to take into account this case, we devised another algorithm called **node-intersection**. In this case, the union of all edge recommendations per tag is used when computing the PageRank, but only those *nodes* involved in recommendations for *all tags* are kept (hence, *node-intersection*). This is another possible way to obtain a subgraph having only a specific facet-related information.

In the following paragraphs, we describe several efficient algorithms for the online computation of facet-dependent rankings.

Single-ranking. A simple online faceted ranking consists of a monolithic ranking of the full graph, without any consideration of tags, which is then filtered to exclude those nodes that are not related to all tags in the facet.

Winners-intersection. In this case, as well as in the next two algorithms, the *offline phase* consists of running PageRank on each tag-related subgraph (such as $G(music)$) and storing only the best-w ranked users. The choice of an adequate number w is application-dependent. For this paper, we have arbitrarily chosen $w = 128$. We shall show that reasonably good results are obtained even for this small value of w. Given a conjunction-of-tags facet, a new graph is constructed by considering only the w "winners" corresponding to each tag and the edges connecting them. A facet-related ranking is then calculated by means of the PageRank algorithm applied to this reduced graph.

[4] We have observed that as the graph grows the relative frequency of tags usage converges. Similar behavior was observed for particular resources by other authors (Golder and Huberman, 2006).

Probability-product. Let us recall that PageRank is based on the idea of a random web-surfer and nodes are ranked according to the estimated stationary probability of such a surfer being at each node at any given time. This basic concept together with the product rule for the joint probability of independent events motivated the **probability-product** algorithm. This algorithm pre-computes ranking for each tag-related subgraph. A ranking associated with a conjunction-of-tags facet is then calculated *online* by multiplying, on a node-by-node basis, the probabilities corresponding to each tag in the facet.

Rank-sum. Consider a recommendation graph G larger than that in Fig. 1 and the query $blues \wedge jazz$. Assume that the PageRank of the top three nodes in the rankings corresponding the subgraphs $G(blues)$ and $G(jazz)$ are as given in Table 3. Ignoring other nodes, the ranking given by the **probability-product** rule is A, B and C. However, it may be argued that node B shows a better equilibrium of PageRank values than node A. Intuitively, one may feel inclined to rank B over S given the values in the table. In order to follow this intuition, we devised the **rank-sum** algorithm which is also intended to avoid *topic drift* within a queried facet, that is, any tag prevailing over the others. Given a conjunction-of-tags facet, the **rank-sum** algorithm adds-up the ranking position of nodes in each tag-related subgraph. The corresponding facet-related ranking is calculated by ordering the resulting sums (see Table 3).

Table 3: **Probability-product** vs. **rank-sum** in an example.

Node	$G(blues)$	$G(jazz)$	Prob.-pr.	Rank-sum
A	0.75	0.04	0.03	4
B	0.1	0.1	0.01	3
C	0.01	0.05	0.005	5

The first two columns show the probability of each node according to PageRank on the corresponding tag-related subgraph.

4.1 Scalability Analysis

As noticed in (Langville and Meyer, 2003), the number of iterations of PageRank is fixed when the tolerated error and other parameters are fixed. As each iteration consists of a sparse adjacency matrix multiplication, the time complexity of PageRank is linear on the number of edges of the graph. Since **probability-product**, **rank-sum** and **winners-intersection** algorithms require the offline computation of PageRank for each tag-related subgraph, it is clear that, if the average number of tags per edge is constant or grows very slowly as the graph grows, then the *offline phase* of these algorithms is scalable, i.e., linear on the number of edges of the complete tagged-graph. Fig. 3 shows that the distribution of tags per edge falls quickly, having a mean of 9.26 tags per edge for the *YouTube* tagged-graph and 13.37 for the *Flickr* tagged-graph. These are *not* heavy-tailed distributions and, since tags are manually added to each uploaded content, we do not expect the average number of tags per recommendation to increase significantly as the graph grows. In other words, Fig. 3 validates the hypothesis on which the scalability of the *offline phase* of the **probability-product**, **rank-sum** and **winners-intersection** algorithms is based.

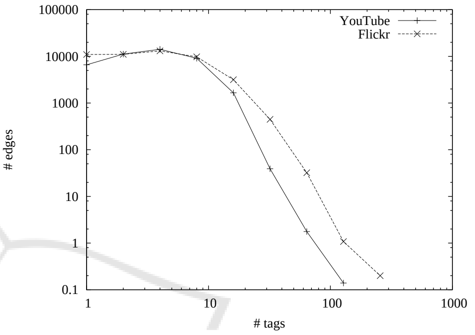

Figure 3: The distribution of number of tags per edge.

The time complexity of the **edge-intersection** algorithm can be estimated by decomposing it into three phases. The first step is to find the graph associated to a facet with k tags, which takes $\mathcal{O}(k \cdot E_{tag})$, where E_{tag} is the number of edges of the largest tag-related subgraph. The second step is to apply the PageRank algorithm to the resulting graph, taking $\mathcal{O}(E_{facet})$ time, where E_{facet} is the number of edges of the graph and, clearly, $E_{facet} \leq k \cdot E_{tag}$. Finally, the list of N_{facet} nodes of the graph must be ordered, taking $\mathcal{O}(N_{facet} \log(N_{facet}))$. We have found that N_{facet} is, in general, much smaller than E_{facet} (see Tables 1-2).

For the **node-intersection** algorithm, the time complexity is the same that in the former case, but E_{facet} and N_{facet} are usually larger because the associated facet-dependent graph includes more edges.

In the case of **single-ranking**, the *online* computation takes $\mathcal{O}(k \cdot N_{tag})$ time, where N_{tag} is the maximum number nodes of a tag-related subgraph. Indeed, assuming that there is an ordered list of nodes related to each tag, its (ordered) intersection can be computed in time that grows linearly with the sum of the lengths of the lists.

The **winners-intersection**, **probability-product** and **rank-sum** algorithms have the same time complexity, $\mathcal{O}(k)$, because they only consider the top-w

ranked users of each tag in the facet and, hence, their complexity depends only on the number of tags (i.e, the number of operations for each tag is fixed by the constant *w*).

5 EXPERIMENTAL RESULTS

In this section, we compare the behavior of the algorithms presented in Sect. 4. As a basis of comparison we use two algorithms whose online computation is unfeasible, but which are reasonably good standards: **edge-intersection** and **node-intersection**.

In order to quantify the "distance" between the results given by two different algorithms, we use two ranking similarity measures, *OSim* (Haveliwala, 2002) and *KSim* (Kendall, 1938; Haveliwala, 2002). The first measure, $OSim(R_1, R_2)$ indicates the degree of overlap between the top *n* elements of rankings R_1 and R_2, i.e., the number of common nodes. The second measure, $KSim(R_1, R_2)$ is a variant of Kendall's distance that considers the relative orderings, i.e., counts how many inversions are in a fixed top set. In both cases, values closer to 0 mean that the results are not similar and closer to 1 mean the opposite.

We have run our algorithms on all facets of tag pairs extracted from the 100 most used tags[5] in each of the graphs, *YouTube* and *Flickr*. For each tag pair, the proposed merging algorithms (**single-ranking**, **probability-product**, **rank-sum** and **winners-intersection**) were compared to the reference algorithms (**edge-intersection** and **node-intersection**) using *OSim* and *KSim* to measure the rankings' similarity.

Tables 4-5 present a summary of the comparisons, where we display averaged similarities for different sizes of top-sets of ranked users. Figures 4 and 5 show a more detailed summary of results for the *OSim* metric in the case of the *YouTube* graph. We do not include gray-scale figures corresponding to *Flickr* because they are very similar. The *x*-axis in the figures corresponds to the number of nodes resulting from the basis of comparison algorithm (**edge-intersection** or **node-intersection**) and the *y*-axis to the top number *n* of nodes used to compute the similarities. The similarity results of *OSim* were averaged for log-log ranges. Observe that darker tones correspond to values closer to 1, i.e., more similar results. White spaces correspond to cases for which there are no data, e.g., whenever the *y* coordinate is greater than intersection size.

[5] Some tags like *you*, *video* or *youtube* which give no information were removed from the experiment.

Table 4: *YouTube*: Comparison of ranking algorithms.

Average similarity to edge-intersection

Algorithm	OSim|KSim		
	top 8	top 16	top 32
Single-ranking	0.08|0.48	0.10|0.50	0.13|0.51
Winners-inters.	0.06|0.48	0.04|0.49	0.04|0.50
Prob.-product	**0.72**|**0.71**	**0.80**|**0.78**	**0.86**|**0.83**
Rank-sum	**0.73**|**0.72**	**0.81**|**0.79**	**0.86**|**0.84**

Average similarity to node-intersection

Algorithm	OSim|KSim		
	top 8	top 16	top 32
Single-ranking	0.31|0.53	0.34|0.55	0.39|0.56
Winners-inters.	0.10|0.49	0.08|0.50	0.08|0.51
Prob.-product	**0.42**|**0.59**	**0.52**|**0.66**	**0.67**|**0.74**
Rank-sum	**0.41**|**0.58**	**0.50**|**0.64**	**0.67**|**0.72**

Table 5: *Flickr*: Comparison of ranking algorithms.

Average similarity to edge-intersection

Algorithm	OSim|KSim		
	top 8	top 16	top 32
Single-ranking	0.07|0.48	0.09|0.49	0.11|0.50
Winners-inters.	0.30|0.53	0.23|0.53	0.11|0.51
Prob.-product	**0.57**|**0.63**	**0.64**|**0.66**	0.72|**0.71**
Rank-sum	**0.57**|**0.63**	**0.64**|**0.67**	**0.72**|**0.72**

Average similarity to node-intersection

Algorithm	OSim|KSim		
	top 8	top 16	top 32
Single-ranking	0.17|0.50	0.21|0.51	0.27|0.53
Winners-inters.	0.19|0.50	0.19|0.52	0.18|0.53
Prob.-product	**0.32**|**0.55**	**0.42**|**0.59**	**0.56**|**0.67**
Rank-sum	**0.31**|**0.53**	**0.41**|**0.58**	**0.56**|**0.66**

5.1 Discussion

As can be appreciated from Tables 4-5 and Figures 4-5, the **single-ranking** algorithm gave the worst results in most cases.

The **winners-intersection** algorithm, which is based on retaining only the 128 top-ranked users for each tag, gives worse results than **probability-product** and **rank-sum**, even for smaller intersections. This fact is explained by the relevance of a large number of recommendations of low-ranked users when computing the PageRank in both the **edge-intersection** and the **node-intersection** cases. Also note that the **winners-intersection** approach gave better results for *Flickr* than for *YouTube*. A possible cause is the assortativeness of *Flickr*'s graph (see Sect. 3.1). Indeed, since assortativeness implies that users with many recommendations are preferentially recommended by users with also many recommendations, the relevance of low-ranked users in the computation of the centrality measure is lower.

The **probability-product** and **rank-sum** algorithms exhibit a similar behavior and clearly outperform other ranking algorithms when considering the similarity to the **edge-intersection** and the **node-intersection** standards.

We should note that we have run experiments with larger values of *w*, the number of "winners" which are stored for each tag, but the behavior of the algorithms was similar.

6 SUMMARY

We have proposed different algorithms for merging tag-related rankings into complete faceted-rankings of users in collaborative tagging systems. In particular, two of our algorithms, **probability-product** and **rank-sum** are feasible for online computation and give results comparable to those of two reasonable, though computationally costly, standards.

A prototypic application which uses the **rank-sum** and the **probability-product** algorithms, is available online (Egg-O-Matic, 2008).

A matter of future research is the possibility of reducing the the complexity of the proposed algorithms by first clustering the tags into topics of interest as done by (Li et al., 2008).

This work also opens the path for a more complex comparison of reputations, for example by integrating the best positions of a user even if the tags involved are not related (*disjunctive* queries) in order to summarize the relevance of a user generating content on the web. It is also possible to extend the algorithms in Sect. 4 to merge rankings generated from different systems (*cross-system ranking*).

REFERENCES

Abel, F., Henze, N., and Krause, D. (2008). A Novel Approach to Social Tagging: GroupMe! In *the 4th WEBIST*.

Al-Khalifa, H. S., Davis, H. C., and Gilbert, L. (2007). Creating structure from disorder: using folksonomies to create semantic metadata. In *the 3rd WEBIST*.

Al-Saffar, S. and Heileman, G. (2007). Experimental Bounds on the Usefulness of Personalized and Topic-Sensitive PageRank. In *Proc. of the IEEE/WIC/ACM International Conference on Web Intelligence*, pages 671–675.

Christopher (2008). *Introduction to Information Retrieval*. Cambridge University Press.

DeLong, C., Mane, S., and Srivastava, J. (2006). Concept-Aware Ranking: Teaching an Old Graph New Moves. *icdmw*, 0:80–88.

Egg-O-Matic (2008). http://egg-o-matic.itba.edu.ar/.

Eirinaki, M. and Vazirgiannis, M. (2005). Usage-Based PageRank for Web Personalization. In *ICDM '05: Proc. of the Fifth IEEE International Conference on Data Mining*, pages 130–137.

Golder, S. and Huberman, B. A. (2006). Usage patterns of collaborative tagging systems. *Journal of Information Science*, 32(2):198–208.

Guo, Y. Z., Ramamohanarao, K., and Park, L. A. F. (2007). Personalized PageRank for Web Page Prediction Based on Access Time-Length and Frequency. In *Proc. of the IEEE/WIC/ACM International Conference on Web Intelligence*, pages 687–690.

Haveliwala, T. H. (2002). Topic-sensitive PageRank. In *Proc. of the Eleventh International World Wide Web Conference*.

Hotho, A., Jäschke, R., Schmitz, C., and Stumme, G. (2006). Information Retrieval in Folksonomies: Search and Ranking. In *Proc. of the 3rd European Semantic Web Conference*, volume 4011 of *LNCS*, pages 411–426.

Jeh, G. and Widom, J. (2002). Scaling personalized web search. Technical report, Stanford University.

Kendall, M. G. (1938). A New Measure of Rank Correlation. *Biometrika*, 30(1/2):81–93.

Langville, A. N. and Meyer, C. D. (2003). Survey: Deeper Inside PageRank. *Internet Mathematics*, 1(3).

Li, X., Guo, L., and Zhao, Y. E. (2008). Tag-based social interest discovery. pages 675–684. WWW.

Marlow, C., Naaman, M., Boyd, D., and Davis, M. (2006). HT06, tagging paper, taxonomy, Flickr, academic article, to read. In *Proc. of the seventeenth conference on Hypertext and hypermedia*, pages 31–40.

Newman, M. E. J. (2002). Assortative Mixing in Networks. *Phys. Rev. Lett.*, 89(20):208701.

Page, L., Brin, S., Motwani, R., and Winograd, T. (1998). The PageRank Citation Ranking: Bringing Order to the Web. Technical report, Stanford Digital Library Technologies Project.

Pastor-Satorras, R. and Vespignani, A. (2004). *Evolution and structure of the Internet: A statistical physics approach*. Cambridge University Press.

Richardson, M. and Domingos, P. (2002). The Intelligent Surfer: Probabilistic Combination of Link and Content Information in PageRank. In *Advances in Neural Information Processing Systems 14*. MIT Press.

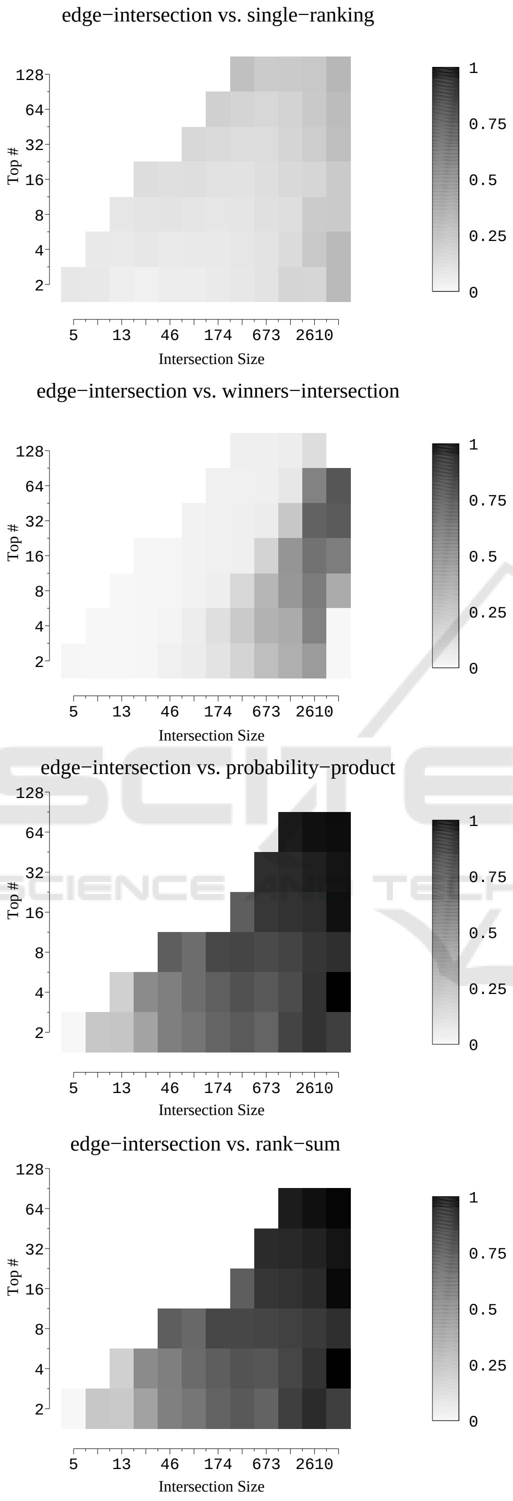

Figure 4: Average similarity (*OSim*) to **edge-intersection** in *YouTube*.

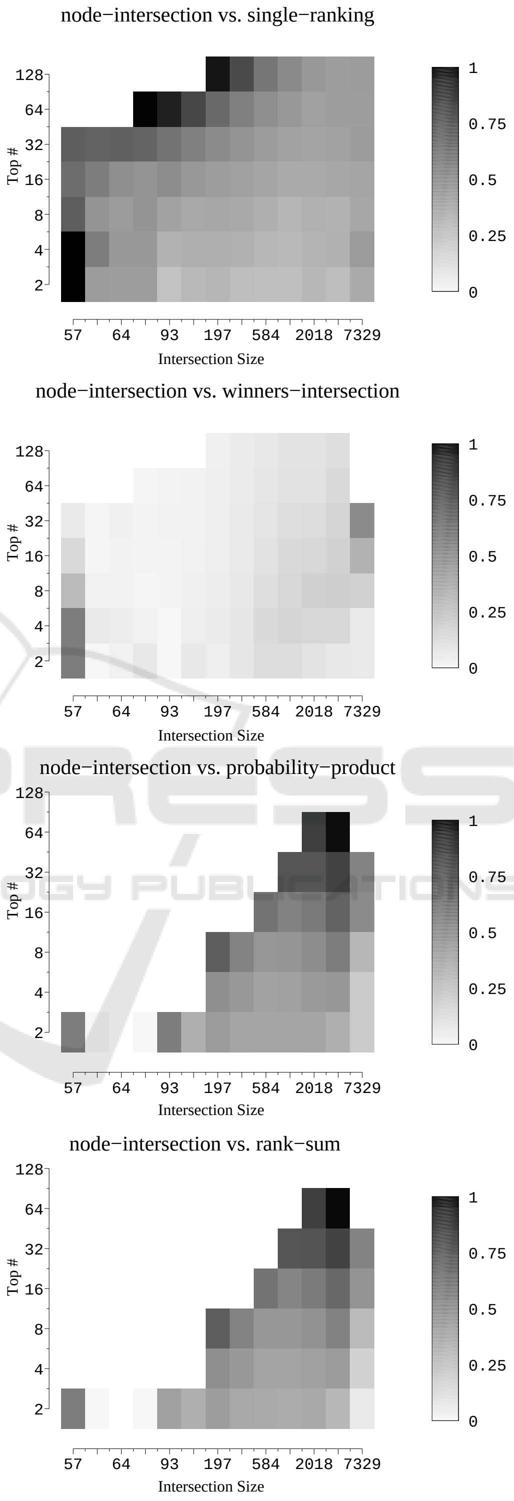

Figure 5: Average similarity (*OSim*) to **node-intersection** in *YouTube*.