Challenges in Personalized Authority Flow Based Ranking of Social Media

Hassan Sayyadi
University of Maryland
sayyadi@cs.umd.edu

John Edmonds
University of Maryland
jedmond3@umd.edu

Vagelis Hristidis
Florida International University
vagelis@cis.fiu.edu

Louiqa Raschid
University of Maryland
louiqa@umiacs.umd.edu

ABSTRACT

As the social interaction of Internet users increases, so does the need to effectively rank social media. We study the challenges of personalized ranking of blog posts. Web search techniques are inadequate since social media lack many of the characteristics of the Web such as rich document content and an extensive hyperlink graph. Further, user behavior in social media has moved beyond keyword based search and must support users who follow a particular blog or theme. In this research, we extend a social media dataset to exploit the associations between authors, blog posts, and categories (topics) of the posts. We then apply personalized authority flow based ranking algorithms based on the random surfer model. We evaluate our personalization approaches through an extensive study on a range of virtual users whose preferences are defined based on intuitive criteria. Our evaluation shows that the accuracy of our personalized recommendations ranges from good to very good for a majority of users, and outperforms reasonable baseline approaches.

1. INTRODUCTION

Social media and the social interactions of users on the Internet have the potential to become a vital source of breaking news as well as knowledge reflecting the expertise of the crowds. The importance of such data has been acknowledged by Web search engines, which now index blog sites. Such sources are of particular importance in evolving situations such as disasters or other unplanned events where the most knowledgeable experts may not be known a priori; there is a need for a diversity of information; information evolves over time and the quality of information can vary. These factors increase the value of social media and social interactions as a valuable source of information.

On the other hand, crowd sourcing can also create a massive stream of irrelevant and low quality information. For instance, users of social networking sites (LinkedIn, Twitter) may receive hundreds or thousands of daily blogs, messages, forums, etc., from other users and subscribed groups. The challenge is to benefit from the potentially valuable nuggets of social media, while mitigating information overload. We propose solutions to effectively rank blog postings in a personalized way, by analyzing their content to discover key topics and exploiting their explicit categorization and author information.

Current Web search and personalization techniques cannot be directly applied since social media typically do not provide the rich document content or structure of Web pages. They also do not provide the complex hypergraph of the Web that is critical for both ranking and personalization. For example, a key principle in Web ranking is that the pages of a good domain are uniformly good. This principle is hard to apply to blogs since a blog domain may host thousands of blogs of diverse quality and importance.

Further, user behavior in social media is different from search behavior on the Web. As discussed in [12], users submit ad hoc keyword queries on the Web. In contrast, social media users may follow posts about a particular topic, or they may follow their favorite author or category. Hence, the topics and authors that a user has liked in the past provide valuable information for the personalized ranking of future blog posts. Identifying the topic of a blog post is challenging. In this work we consider the expressions of the topic by including explicit category names for blogs.

We have several objectives in this research. First, we enhance a blog dataset so that we may apply authority flow based ranking. Such rankings have been shown to be effective to rank entities on the Web (PageRank [14]) and in structured databases (ObjectRank [7]). We add nodes, e.g., authors, category, etc., and edges to the existing dataset. This leads to a form of entity-relationship graph which conforms to an entity-relationship schema.

An example entity-relationship schema to facilitate authority flow ranking includes four entity types, *BlogPost*, *Author*, and *Category* (explicit topic of a post), as well as the corresponding edge types, as shown in Figure 1.

The second objective is to develop a suite of authority flow-based personalized ranking techniques based on the random surfer model. We consider three personalization techniques, including Personalized PageRank (pPR), ObjectRank (OR), and an information retrieval approach using Apache Lucene (pIR).

Our next objective is an experimental study on the accuracy of personalized recommendations. We conduct experiments on a slice of the Spinn3r dataset [3]. To study effectiveness, we performed an experiment with various types of *virtual users* where each user has a profile represented by a personalized ranking of their daily favorite *BlogPost* entries. In particular, we consider virtual users who follow the posts of a set of bloggers (*Author Users*), or the posts of a set of categories (*Category Users*), or the posts related to a set of keywords (*Keyword Users*).

Our experiments reflect that pOR can provide accurate personal-

ization for the majority of virtual users. We use the F_1-score to compare the effectiveness of the various ranking techniques, which combines the precision and the recall measures.
In summary, our paper makes the following contributions:

- We enhance the social media dataset so that we can apply authority flow based ranking. Further, we exploit massively parallel document similarity techniques, using the MapReduce paradigm, to measure the similarity between BlogPosts[9]. (Section 3)
- We present a suite of novel and baseline ranking techniques for BlogPosts. (Section 4)
- We evaluate the effectiveness of the proposed ranking techniques using various type of virtual users. We demonstrate that pPR and pOR can significantly outperform the baseline pIR. Further, pOR outperforms pPR. (Section 5)

We present related work in Section 2 and we conclude in Section 6.

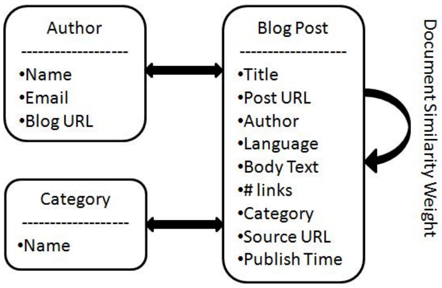

Figure 1: Enhanced Social Media Schema Graph

2. RELATED WORK

The TREC Blog Track [11, 13] has taken the lead to identify several challenges for social media. This includes the following: topic detection and tracking (TDT) (to cluster documents around topics or events); the blog distillation task (BDT) that determines authoritative sources for some topic X; the faceted blog distillation task to identify quality cues such as opinion; and the top news stories tasks that uses blog posts as a proxy for breaking news.

Personalized ranking has been proposed for some time now, especially in the context of Web search. For instance, while PageRank (PR) [14] gives a global ranking for all the Webpages, users frequently have different points of views. Personalized PageRank (pPR) produces an ordered list for the user which should reflect the user's preference or profile. Major commercial search engines provide personalization by accommodating the topics of interests, prior search history, or other descriptions of users' preferences. A major approach to personalization is based on authority flow-based ranking [6, 15, 2], which we discuss in detail in the next section.

Research in [1] analyzes Weblogs to infer paths that reflect the propagation of information. This inference includes a novel utilization of historical data to identify repeating patterns. iRank is a ranking algorithm for blogs; a graph is created by adding an edge from a blog that first mentions a topic to a blog that later mentions the same topic. iRank executes PageRank to find the most *influential* blogs. In our work, we also create implicit edges based on similarity between the blogs, but our focus is on personalization and not on a global ranking.

The EigenRumor algorithm [5] creates a bipartite graph of authors and blogs, where edges are added between the author of a blog and her blog, and between an evaluator of a blog and the blog. However, this last type of edge is hard to capture and quantify in practice. Then, EigenRumor applies the HITS algorithm [8]. Corso et al. [4] rank a stream of news by applying authority flow techniques on a graph that contains news sources and articles. News sources are linked to their articles and articles are connected to each other basd on text similarity. While all of these approaches enhance the social media dataset, none of their approaches can be applied to our task of personalized recommendation.

3. SOCIAL MEDIA DATASET

The dataset provided by Spinn3r.com, is a set of 44 million blog posts made between August 1st and October 1st, 2008 [3]. The post includes the text as well as metadata such as the blog's homepage, timestamp, etc. The data is formatted in XML and is further arranged into tiers approximating search engine ranking. We first discuss enriching the social media schema graph and dataset preparation.

Schema Graph for personalized PR (pPR) and ObjectRank (pOR) A shortcoming of social media is the lack of a rich hypergraph to determine the importance of pages. Following the example of projects on ranking collections of documents that are unconnected by hyperlinks, we first compute the pairwise document similarity between two *BlogPost*s using Cloud Computing Framework[9] and insert a reflexive edge between two entries as seen in Figure 1; the label *doc-sim-weight* reflects the document similarity value for each edge.

Next, to reflect authority flow, we utilize an entity-relationship schema with three entity types, *BlogPost*, *Author*, and *Category* (topic of the post), as shown in Figure 1. The concepts of *BlogPost*, *Author* and *Category* are intuitive to understand and these nodes are easily identified in social media datasets. There are also edges to represent the associations between *BlogPost* and *Category*, *BlogPost* and *Author*, and the reflexive edge from *BlogPost* to *BlogPost* representing document similarity.

Data Graph We use a 31 day slice of the data (August 2008) to create data graphs for both training and testing. After removing non-English posts and posts without Author or Category information, we get a subset of data which contains approximately 800,000 posts.

Next, we create a data graph that is appropriate for personalized ranking. This requires that we filter the dataset so that the distribution of posts per author, or posts per category, or categories per post, reflect a non sparse and normalized distribution. Further, we need sufficient data for both training and testing. For example, if we wish to consider a *virtual user* who is following a particular author, and we request at least H posts for training, then we must restrict our dataset to authors who have more than H posts. We note that many posts are not labeled with category labels. In addition, category labels can be inconsistent and sparse, since the category labels are chosen arbitrarily by authors. Hence, some category labels are only used by one author or a few authors.

We use the following procedure to create a data graph for personalization: We identified frequent categories (at least 50 posts per category) and frequent authors who had at least 10 posts from the frequent categories. Then, we selected posts written by these frequent authors and labeled with these frequent categories. Our experiment dataset comprises a data graph of 248908 nodes (including 137047 *BlogPosts*,2210 *frequent Authors*, and 109651 *frequent Categories*) and 1391467 edges (including 137047 Author_BlogPost edges, 794005 Category_BlogPost edges, and 582192 BlogPost_BlogPost edges).

4. PERSONALIZED RANKING

ObjectRank [2] personalizes ranking in Entity-Relationship graphs; it models nodes as entity types and groups edges by their edge type or semantic type. Then, the authority flow is personalized by a *weight assignment vector* (WAV) Θ for each edge type. The WAV determines the importance of each type of association in the ranking. For example, in Figure 1, there are 3 edge types associating *BlogPost* to *BlogPost*, to *Author*, etc. The WAV for this graph has 6 entries, one for each direction of an edge type. Varadarajan et al. [16] present techniques to learn the WAV using relevance feedback.

We consider the following personalization approaches on the enriched social media graph of Figure 1.

pPR: We apply personalized PageRank (pPR) on a restricted graph that has one entity type *BlogPost* and a reflexive edge from *BlogPost* to *BlogPost* with a document similarity edge weight. There will also be a personalized base set of *BlogPost* entries. Details of computing document similarity and the choice of queries and personalized base set for evaluation are discussed later. This variant pPR will be the baseline for our evaluation of personalized ranking.

pOR: Personalized ObjectRank (OR) will be evaluated on the entity-relationship schema graph of Figure 1. For this variant, we will choose a *default WAV*, i.e., the values in Θ are equal weights for all type of edges for each type of node. Thus, for this variant of pOR, we are determining the impact of only enriching the social media schema graph, but not using personalized values for Θ.

pOR will also use a personalized base set of *BlogPost* entries.

pIR: We use an extension of the Apache Lucene text search engine [10]. Personalization is implemented by using all the keywords in the personalized baseset of each BlogPost entry to create a document query [17]. This will also serve as a baseline.

In this paper we manually assign appropriate values for WAV Θ, based on the specific type of virtual user. Specific values of Θ are given in the evaluation section. In related research, we have used relevance feedback from human users to learn the appropriate values for WAV Θ [16]. However, we do not apply such techniques in this research.

5. EVALUATION

We present the results of an experimental evaluation of the accuracy of personalized authority flow based ranking using the enhanced Spinn3r dataset. In the first set of experiments, we consider several classes of virtual users and we determine the accuracy of pIR, pPR and pOR for these virtual users. Then, we examine the sensitivity of pOR for various parameters.

We consider three classes of virtual users. For each class of virtual user, we identify the corresponding `ground truth` of posts that are relevant to the user. The ground truth will be partitioned into a *personalized training base set* and a *testing set*. **Author Users:** These users are interested in posts by some specific authors; both the *personalized training base set* and *testing set* are posts by a selected author. **Category Users:** These users are interested in posts labeled with specific categories; both the *training* and *testing* posts are posts that are labeled with a specific category. **Keyword Users:** These users are interested in posts that are most relevant to some set of keywords. All *BlogPost*s were indexed using the Apache Lucene [10] text search engine and the Top K=100 posts for each keyword was retrieved to serve as the ground truth.

We consider several parameters to test the sensitivity and robustness of the personalization variants. For each virtual user experiment we vary three parameters as follows: The first parameter D is the number of distinct queries for a virtual user. For example for D =2, an Author User will follow the posts of 2 authors, or a Category User will follow posts labeled with 2 Category keywords. The second parameter H reflects the cardinality of the *personalized training base set*. The set of ground truth *BlogPost*s will be sorted in chronological order and partitioned into two parts; the first H posts are for the *personalized training base set*. The third parameter U indicates the number of virtual users that are generated for each experiment. Results are reported as an average over U users.

For each experiment, the *personalized training base set* is provided to pOR. The default values for the WAV Θ are equal weights for all type of edges for each type of nodes. After training, pOR then returns a set of personalized recommendations, *pRec*. The cardinality of the *pRec* set is chosen to be the cardinality of the ground truth *testing set*.

We note that for Author and Category Users, one can only determine if a recommended post in *pRec* is contained in the ground truth, and is relevant to the user, or if it is not contained in the ground truth, and thus is not relevant to the user. For the Keyword Users, one could utilize the ranking provided by the search engine. However, for uniform reporting of experiment results, we use the binary result of relevant or not relevant, if the recommended post in *pRec* is contained in the ground truth.

To measure the accuracy of the personalized ranking, we report on the $F1$ measure for *pRec*, averaged over the U users. $F1$ is the harmonic mean of precision and recall; it has a best case score of 1.0 and a worst case score of 0. Recall that the cardinality of *pRec* is identical to the cardinality of the ground truth *testing set*. Hence, for our experiments, the value of $F1$ for *pRec* is equal to both the precision and the recall.

Comparison of Ranking Algorithm Variants We compare the performance of the ranking variants for all types of users as follows

pIR: we view each post as a document that concatenates the posts' text, categories, and author. The user's base set of posts is concatenated into a single user profile document P. We rank the candidate posts by their IR similarity to P, where the similarity is computed using Lucene[10]. This variant is one of the baslines for comparison.

pPR: We use the schema graph of Figure 1 restricted to one node type, *BlogPost*, and one reflexive document similarity link. This variant is a second baseline for comparison.

pOR: We use the schema graph of Figure 1 including all nodes and edges and using equal weights for edges.

Figure 2 shows the improvement of pOR, in comparison to the authority flow baseline of pPR and Full text search baseline of pIR, for Author Users, Category Users, and Keyword Users. The most significant improvement was observed for Author Users. Surprisingly, pIR is not performing well for Keyword users. The reason is that the user profile document P is too big, so the weight of the user's profile keywords is not high.

Using the Wilcoxon Signed Rank test for statistical significance we find that the F1 score for pOR significantly dominates pPR and pPR significantly dominates pIR for author and category users at the 95 confidence level. Furthermore, the F1 score for pOR and pPR significantly dominate pIR for keyword at the 95 confidence level.

Parameter Sensitivity Analysis

Next, we report on the sensitivity and robustness of the personal-

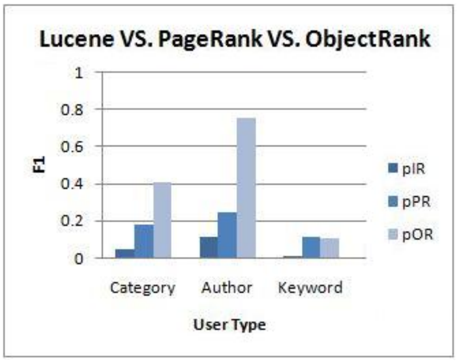

Figure 2: $F1$ Values of pIR and PageRank versus ObjectRank for Author and Category users(U=50, D=6, H=30), and Keyword Users (U=10, D=1, H=30).

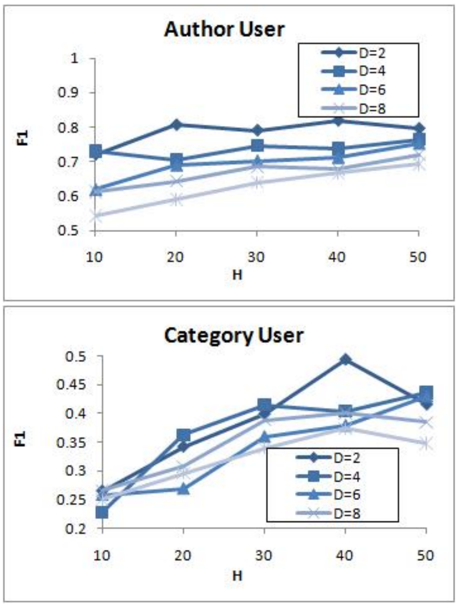

Figure 3: Average $F1$ Values of pOR for 50 Author and Category Users ($U = 50$) with Varying Values of $D = 2, 4, 6, 8, 10$ and $H = 10, 20, 30, 40, 50$.

ization variants. Recall that we varied three parameters as follows: D - the number of distinct queries for a virtual user; H i-the cardinality of the *personalized training base set*; U - the number of virtual users for each experiment.

Figure 3 reports on the value of $F1$ for Author and Category Users. The X axis varies the value of H from 10 to 50. Each of the curves in the Figure corresponds to values of D varying from 2 to 10. Each data point in each plot is averaged over $U = 50$ Author Users. As observed in the Figure, the value of $F1$ is highest for lower values of $D = 2$ compared to higher values of $D = 10$; this is because larger values of D represent a diversity of queries for each Author/Category User. As expected, the $F1$ value increases as the value of H increases, or the cardinality of the *personalized training base set* increases. This benefit from increasing H values is observed clearly for the case of $D = 10$. As the value of H approaches 50, the $F1$ values converge, indicating there is no additional benefit of higher H values.

6. CONCLUSIONS

We extended a social media dataset and provided accurate personalized authority flow based ranking for various type of virtual users. We presented a suite of blogs ranking techniques, which we experimentally compared. In the future, we will also model more sophisticated virtual users to better capture real users' behavior as well as doing experiments with real users. We will also experiment pOR+ for real users that uses relevance feedback to learn the best edge weights for Θ.

7. ACKNOWLEDGMENTS

This research was partially supported by NSF awards CMMI 0753124 and IIS 0960963. Vagelis Hristidis was partly supported by NSF grants IIS-0811922 and IIS-0952347, and DHS grant 2009-ST-062-000016.

8. REFERENCES

[1] E. Adar, L. Zhang, L. Adamic, and R. Lukose. Implicit structure and the dynamics of blogspace. In *Proceedings of the Workshop on the Weblogging Ecosystem in conjunction with WWW2004*, 2004.

[2] A. Balmin, V. Hristidis, and Y. Papakonstantinou. Objectrank: Authority-based keyword search in databases. In *VLDB*, pages 564–575, 2004.

[3] K. Burton, A. Java, and I. Soboroff. The icwsm 2009 spinn3r dataset. In *Proceedings of the Conference on Weblogs and Social Media (ICWSM 2009)*, 2009.

[4] G. M. Del Corso, A. Gullí, and F. Romani. Ranking a stream of news. In *WWW '05: Proceedings of the 14th international conference on World Wide Web*, 2005.

[5] K. Fujimura, T. Inoue, and M. Sugisaki. The eigenrumor algorithm for ranking blogs. In *Proceedings of the Workshop on the Weblogging Ecosystem*, 2005.

[6] T. H. Haveliwala. Topic-sensitive pagerank. In *WWW '02*.

[7] V. Hristidis, H. Hwang, and Y. Papakonstantinou. Authority-based keyword search in databases. *ACM Trans. Database Syst.*, 33(1):1–40, 2008.

[8] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, 1999.

[9] J. Lin. Brute force and indexed approaches to pairwise document similarity comparisons with mapreduce. 2009.

[10] Lucene. http://lucene.apache.org/java/docs/.

[11] C. Mcdonald, I. Ounis, and I. Soboroff. Overview of the trec 2009 blog track. 2009.

[12] G. Mishne and M. de Rijke. A study of blog search. *Lecture Notes in Computer Science*, 3936/2006, 2006.

[13] I. Ounis, C. Macdonald, and I. Soboroff. Overview of the trec-2008 blog track. In *Proceedings of the Text REtrieval Conference (TREC)*, 2008.

[14] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. In *Stanford Tech Report*, 1998.

[15] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[16] R. Varadarajan, V. Hristidis, and L. Raschid. Explaining and reformulating authority flow queries. In *ICDE '08*.

[17] Y. Yang, N. Bansal, W. Dakka, P. G. Ipeirotis, N. Koudas, and D. Papadias. Query by document. In *WSDM*, pages 34–43, 2009.