Blog Distillation using Random Walks

Mostafa Keikha
Faculty of Informatics
University of Lugano
Switzerland
keikham@lu.unisi.ch

Mark James Carman
Faculty of Informatics
University of Lugano
Switzerland
mark.carman@lu.unisi.ch

Fabio Crestani
Faculty of Informatics
University of Lugano
Switzerland
fabio.crestani@unisi.ch

ABSTRACT

This paper addresses the blog distillation problem. That is, given a user query find the blogs most related to the query topic. We model the blogosphere as a single graph that includes extra information besides the content of the posts. By performing a random walk on this graph we extract most relevant blogs for each query. Our experiments on the TREC'07 data set show 15% improvement in MAP and 8% improvement in Precision@10 over the Language Modeling baseline.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms

Algorithms

1. INTRODUCTION

Blog search users often wish to identify blogs about a given topic so that they can subscribe to them and read them on a regular basis. Blog distillation is the problem of finding the most relevant blogs for a specific topic. It is different from traditional ad-hoc search since the retrieval unit is a blog (a collection of the posts), not a single document. With this view, it is similar to the task of resource selection in federated search [3]. Other researchers [2] view it as an ad-hoc search task and consider each blog as a long document.

We present a graph based model of the blogosphere that captures relationship between posts, blogs, and terms. By performing a random walk in this graph, we extract the most relevant blogs for a specific query.

2. SMALL DOCUMENT MODEL

We use the small document model as a baseline model to compare our results with. In the small document (SD) model proposed in [3], posts are viewed as documents and blog feed as a collection of these documents. The model considers the relationships among the posts, or between the posts and the blog. The posterior probability of observing a blog given the query is calculated by:

$$P_{SD}(B \mid Q) \overset{rank}{=} \underbrace{P(B)}_{\substack{\text{Blog}\\\text{Prior}}} \sum_{P \in B} \underbrace{P(Q \mid P)}_{\substack{\text{Query}\\\text{Likelihood}}} \underbrace{P(P \mid B)}_{\substack{\text{Post}\\\text{Centrality}}}$$

where P is the post, B is the blog (a collection of posts) and Q is the query. The blog Prior grows logarithmically with the number of posts in the blog, to favor longer blogs. Query likelihood is computed on query terms using Jelinek-Mercer smoothing and the post centrality is computed using the $P(P \mid B) = \prod_{t \in P} P(t \mid B)^{P(t|P)}$, geometric mean of term generation probabilities.

3. RANDOM WALK MODEL

Beside content of the posts there are other sources of information in the blogosphere, such as links between posts, that can be useful in blog distillation. Representing this information in a graph and using link analysis methods is one way to exploit this extra information. Graph based representations have been well studied in Web retrieval [5].

We propose a graph based representation of the blogosphere that includes three types of objects and multiple types of relations between them. Blogs, posts and terms are objects (nodes) in the graph. Figure 1 shows part of such a graph. An edge between a blog and a post shows membership of that post in that specific blog and an edge between a post and a term means the term occurred in that specific post. There is an edge between post i and post j if post i has a hyper-link to post j, or, if they are in the same blog. This edge shows possible similarity between content of the posts. By creating a hyper-link to another post, the author of the post implicitly states that there is a relation between these posts. Posts in the same blog, since they have the same author(s), are very likely to discuss similar topics. The transition matrix for such a graph is:

$$A = \begin{bmatrix} BB & BP & BT \\ PB & PP & PT \\ TB & TP & TT \end{bmatrix} = \begin{bmatrix} 0 & BP & 0 \\ 0 & PP & PT \\ 0 & TP & 0 \end{bmatrix}$$

where B, P, and T are blogs, posts, and terms, respectively. Each element in the matrix A shows a relation between two objects, for example BP is a submatrix in which its elements show relations from blogs to posts. Because there is not any direct relation between some objects, like blogs and terms, those parts of matrix are zero. In this matrix, A_{ij} is the probability of moving from i to j in one step of a random walk and the sum of the probabilities for each row is one.

SIGIR'09, July 19–23, 2009, Boston, Massachusetts, USA.

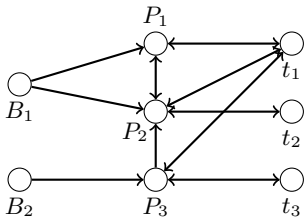

Figure 1: Example of blogosphere graph

Model	MAP	P@10
SD	0.219	0.388
RW	0.253	0.413

Figure 2: MAP and P@10 for implemented models

Thus we can compute the probability of moving between nodes in more than one step by multiplying the matrix by itself. Element c_{ij} in $C = A^n$ (produced by multiplying A by itself n times) is the probability of reaching to j from i after n steps of random walk.

We want to compute $P(t|B_j)$ for each query term. Thus it is more efficient to start from query terms and follow a backward random walk [1].

$$score_{RW}(B_j, Q) = \sum_{t \in Q} P_n(t|B_j) = (A^n M_Q)_j$$

where M_Q is the vector with length N (total number of nodes in the graph), that has the value one for query terms and zero for all other elements. $P_n(t|B_j)$ shows probability of moving from blog B_j to query term t in n steps. An important step in creating this graph is computing the weights of edges. A blog has only one type of relation with its posts. The probability of this relation is computed based on the number of posts in that blog:

$$P(P_i|B_j) = \frac{1}{|B_j|}$$

From one specific post we can have three different outlinks including terms, posts in the same blog or posts related with hyper-links. The sum of all these relations should be one and they can be given different importance.

$$P(t_k|P_i) = \alpha_1 * \frac{tf(t_k, P_i)}{|P_i|}$$

$$P(P_l|P_i) = \alpha_2 * \frac{1}{|Links(P_i)|}, \text{ where } P_i \text{ has a link to } P_l$$

$$P(P_l|P_i) = \alpha_3 * \frac{1}{|B_j| - 1}, \text{ where } P_i,\ P_l \text{ are in the same blog}$$

where $\alpha_1 + \alpha_2 + \alpha_3 = 1$ and are determined experimentally. Weights for outgoing links from terms are computed based on the frequency of the term in the document and in the collection. As such, more frequent terms in the collection will have less weight for their outgoing links:

$$P(P_i|t_k) = \frac{tf(t_k, P_i)}{\sum_{t_k \in P_j} tf(t_k, P_j)}$$

4. EXPERIMENTAL RESULTS

For evaluating our methods we used the TREC Blogs06 test collection and the TREC'07 query sets. This dataset includes the blog posts (permalinks), feeds and homepages for each blog. In our experiments we used only the permalinks component of the collection, which consists of approximately 3.2 million documents [4]. We used the Terrier Information Retrieval system [6] to index the collection and retrieve documents. For each query we select the top 15000 posts by using Terrier version of BM25 (default stemming and stop words). We then add all posts that have inlinks or outlinks to this related set and create a graph for these set of posts. The length of the random walk is set to 25 to be long enough to decrease the effect of query terms and have a better smoothing effect. Table 1 shows MAP and Precision@10 for two implemented models and Figure 3 depicts their Precision-Recall graph.

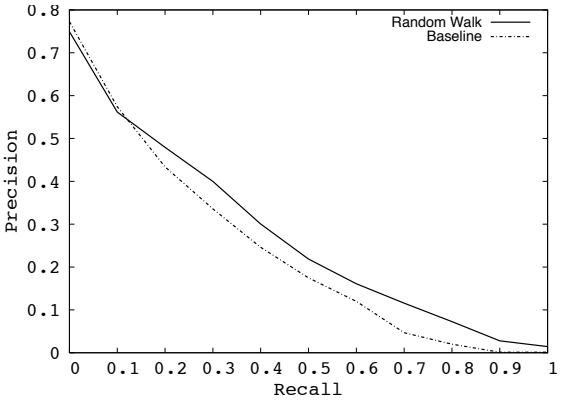

Figure 3: Precision-Recall for implemented models

5. CONCLUSIONS

In this paper we presented a unified graph representation of the blogosphere. This model includes other available information besides contents of the posts, like hyperlinks between posts or co-occurrence of the terms. By using co-occurrence of terms, this model smooths term probabilities on each post. Our experiments show that this method has better results than a baseline small document language model approach.

We have not modelled temporal properties of the posts here. In the future we intend to use this information, for example, by giving higher weights to new posts. Also, we can consider more parameters for defining the weights of edges. Here we gave the same weights to all hyper-links of a post, but some links could be more important based on the destination similarity to the topic or to the source of the link.

6. REFERENCES

[1] N. Craswell and M. Szummer. Random walks on the click graph. In *SIGIR '07*, pages 239–246, New York, NY, USA, 2007. ACM.

[2] M. Efron, D. Turnbull, and C. Ovalle. University of Texas School of Information at TREC 2007. In *Proc. of the 2007 Text Retrieval Conf*, 2007.

[3] J. L. Elsas, J. Arguello, J. Callan, and J. G. Carbonell. Retrieval and feedback models for blog feed search. In *SIGIR*, pages 347–354, 2008.

[4] D. Hannah, C. Macdonald, J. Peng, B. He, and I. Ounis. University of Glasgow at TREC 2007: Experiments in Blog and Enterprise Tracks with Terrier. In *Proceedings of TREC*, 2007.

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

[6] I. Ounis, G. Amati, V. Plachouras, B. He, C. Macdonald, and D. Johnson. Terrier information retrieval platform. In *Proceedings of ECIR'05*, pages 517–519. Springer, 2005.