UNIVERSITY OF AMSTERDAM

UvA-DARE (Digital Academic Repository)

Finding key bloggers, one post at a time

Weerkamp, W.; Balog, K.; de Rijke, M.

Publication date
2008

Published in
Frontiers in Artificial Intelligence and Applications

Link to publication

Citation for published version (APA):
Weerkamp, W., Balog, K., & de Rijke, M. (2008). Finding key bloggers, one post at a time. *Frontiers in Artificial Intelligence and Applications*, *178*, 318-322. http://staff.science.uva.nl/~mdr/Publications/Files/ecai2008.pdf

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, P.O. Box 19185, 1000 GD Amsterdam, The Netherlands. You will be contacted as soon as possible.

UvA-DARE is a service provided by the library of the University of Amsterdam (https://dare.uva.nl)

Download date:08 Oct 2026

Finding Key Bloggers, One Post At A Time

Wouter Weerkamp and **Krisztian Balog** and **Maarten de Rijke**[1]

Abstract. User generated content in general, and blogs in particular, form an interesting and relatively little explored domain for mining knowledge. We address the task of blog distillation: to find blogs that are principally devoted to a given topic, as opposed to blogs that merely happen to discuss the topic in passing. Working in the setting of statistical language modeling, we model the task by aggregating a blogger's blog posts to collect evidence of relevance to the topic and persistence of interest in the topic. This approach achieves state-of-the-art performance. On top of this baseline, we extend our model by incorporating a number of blog-specific features, concerning document structure, social structure, and temporal structure. These blog-specific features yield further improvements.

1 Introduction

With the growth of the blogosphere comes the need to provide effective access to the knowledge and experience contained in the many tens of millions of blogs out there. Information needs in the blogosphere come in many flavors. E.g., Mishne and de Rijke [9] consider both *ad hoc* and *filtering* queries, and argue that blog searches have different intents than typical web searches, suggesting that the primary targets of blog searchers are tracking references to named entities and identifying blogs or posts which focus on a certain concept. The Blogranger system [3] offers several types of search facilities; in addition to post retrieval facilities, it also offers a blog search engine, i.e., an engine aimed at identifing *blogs* about a given topic which a user can then add to an RSS reader.

The task on which we focus in this paper is the *blog distillation* task: to find blogs that are principally devoted to a given topic. That is, instead of identifying individual "utterances" (posts) by bloggers, we want to identify key blogs with a recurring interest in the topic, that provide credible information about the topic. The blog distillation task is an interesting one, for a number of reasons. First, it addresses a real information need, shared by professional and non-professional searchers of the blogosphere. Second, a retrieval model for this task seems to require multiple types of evidence: "local" evidence derived from (a small number of) blog posts of a given blogger plus more "global" evidence derived from a blog as a whole.

Successful approaches at the new feed (blog) distillation task at TREC 2007 Blog track, take the entire blog as indexing unit, that is, the contents of individual posts belonging to the same blog are concatenated into a single document: the blog. Even though this approach performs well in TREC, we want to use individual posts as indexing unit for three (practical) reasons: (i) to allow for easy incremental indexing, (ii) for presentation of retrieval results, posts are natural and coherent units, and (iii) the most important reason, to allow the use of one index for both blog post and blog retrieval.

[1] ISLA, University of Amsterdam, The Netherlands. Email: weerkamp,kbalog,mdr@science.uva.nl

Given this constraint, how, then, should we model the blog distillation task? In this paper we view blog distillation as an association finding task, more specifically, as a blogger-topic association finding task: which blogger is most closely associated with a given topic? Given our choice of working with posts (as opposed to entire blogs) as indexing units, we need effective ways of estimating blogger-topic associations from blog posts. As we will see below, we also need to be able to incorporate the importance of an individual blog post given the blog from which it originates.

Given this choice of model, we explore a number of dimensions. First, how does our (post-based) model perform compared to other known solutions to blog distillation? Second, and assuming that blog distillation is a precision-oriented task (like so many search tasks on the web), can we use the document structure that blogs come with to favor relatively rare but high quality matches; i.e., if we represent blog posts using their titles only (as opposed to title-plus-body) do we observe a strong precision-enhancing effect (perhaps at the expense of recall)? And what if we combine the title-only representation with the title-plus-body representations? Third, how can we make use of blog specific features to improve blog distillation?

Our main finding is that a post-based approach to feed distillation can be as effective as state-of-the-art blog-based approaches. Additionally, on top of this competitive baseline we explore blog-specific features such as document structure, social structure and temporal structure, and find that (i) the lean title-only content representation has a clear precision-enhancing effect when compared to a title+body representation, while a combination of the two representations outperforms both; (ii) post priors based on comments and post length have a positive influence on performance; (iii) temporal ordering of posts can be used as an indication of the importance of a post given a blog; and (iv) a combination of all blog specific features on top of our baseline shows significant improvements over the baseline on most metrics.

The remainder of the paper is organized as follows. In Section 2 we discuss related work. Section 3 details our experimental setup. In Section 4 we introduce our blog distillation model and assess its performance. Next, in Section 5 we discuss the implementation of the blog specific features; this is followed by an experimental evaluation and discussion of our findings in Section 6. We conclude in Section 7.

2 Related Work

Responding to the emerging interest in blogging and in information access tasks for the blogosphere, TREC launched a blog track in 2006 [10]. The first round of this track focused mainly on finding relevant blog *posts*, with a special interest in their opinionatedness. Many insights in blog post retrieval have been gained (see, e.g., [6, 8, 10]), but the task of finding relevant *blogs* has received

less attention.

As part of the TREC 2007 Blog track [6] a new task was introduced: feed distillation. The aim of this task is to return a ranking of blogs rather than individual posts given a topic; this is summarized as *find me a blog with a principle, recurring interest in X*. The scenario underlying this task is that of a user wanting to add feeds of blogs about a certain topic to his or her RSS reader. This task is different from a filtering task [11] in which a user issues a repeating search on posts, constructing a feed from the results.

Prior to TREC 2007 the interest in identifying key blogs was limited. Fujimura et al. [3] propose a multi-faceted blog search engine that allows users to search for blogs and posts. One of the options is to use the blogger filter: the search results (blog posts) are clustered by blog and the user is presented with a list of blogs that contain one or more relevant posts. Ranking of the blogs is done based on the EigenRumor algorithm [2]; in contrast to our method, this algorithm is query-independent.

TREC 2007 witnessed a broad range of approaches to the task of identifying key blogs. Seki et al. [12] experiment with the idea that over time each blog post of a relevant blog should be relevant to the given topic, which they implement in a two-stage retrieval model. A very different approach is tested in [4]; the method uses Web 2.0 applications and thesauri (like Wikipedia, WordNet, Dmoz, etc.) to generate topic maps. After an initial retrieval run against an index consisting of the RSS content (rather than the HTML content) of the blogs, a classifier is used to determine the relevance of blogs regarding the topic (map).

An interesting preprocessing step by Seo and Croft [13] consists of removing all blogs that consist of only one post, since retrieving these blogs would come down to retrieving posts. After this step, three retrieval models are tested: the baseline run builds virtual documents consisting of all posts of a blog and uses a language modeling approach to retrieval of relevant blogs. The weakness of this model is that longer posts may bias the blog's relevance. The second model described in the paper constructs pseudo-clusters based on an initial retrieval run on a post index and ranks blogs using a product over the documents in the cluster. The weakness of this model lies in the fact that a very regularly updated blog (with many posts) is likely to have many posts in the initial post retrieval results, even though this might only be a small portion of the total number of posts in the blog. To counter this, a third model combines the two previous models and uses the first model to penalize blogs that have many non-relevant posts. Results are reasonable, with the combination of models performing best.

The most effective approaches to feed distillation at TREC 2007 were based on using the (aggregated) text of entire blogs as indexing units. E.g., Elsas et al. [1] experiment with a large document model, and a small document model. The former views blogs as a single document, disregarding the fact that a blog is constructed from multiple posts. The latter takes samples of posts from blogs and combines the relevance scores of these posts into a single blog score. Their main outcome is that the large document model outperforms the small document model and that query expansion on Wikipedia is very beneficial (with an increase in MAP scores of almost 10%).

3 Experimental Setup

Before we introduce our modeling of blog distillation in Section 4, we first describe our test collection, the metrics we use, and the smoothing settings we employed.

3.1 Test Collection

As our test collection we use the TRECBlog06 corpus [5]. This corpus has been constructed by monitoring feeds for a period of 11 weeks and downloading all permalinks. For each permalink (or blog post or document) the feed number is registered. Besides the permalinks (HTML documents) we also have syndicated content to our availability; we only used the HTML documents.

For our experiments we construct two indices: a title-only index (T), and a title-and-body index (TB). The former consists of the `<title>` field of the documents, the latter combines this field with the content of the `<body>` part of the documents. Table 1 lists the characteristics of both indices.

index	size	terms	unique terms	avg. length
T	674MB	17.4M	439,747	5
TB	16GB	1,656.3M	9,106,161	515

Table 1. Characteristics of T and TB indices

The TREC 2007 Blog track offers 45 feed distillation topics and assessments [6]. Both topic development and assessments are done by the participants. Assessors were asked to check a substantial number of blog posts of a retrieved feed to determine the relevance of the entire feed.

For all our runs we use the topic field (T) of the topics and ignore all other information available (e.g., description (D) or narrative (N)).

3.2 Metrics

We use the following standard metrics to determine the effectiveness of our retrieval methods: mean average precision (MAP), R-precision (R-prec), as well as three precision-oriented measures: precision at ranks 5 and 10 (P@5 and P@10) and mean reciprocal rank (MRR).

4 Modeling Blog Distillation

To tackle the problem of identifying key blogs given a query, we take a probabilistic approach and formulate the task as follows: *what is the probability of a blog (feed) being a key source given the query topic q?* That is, we determine $p(blog|q)$, and rank blogs according to this probability. Since the query is likely to consist of very few terms to describe the underlying information need, a more accurate estimate can be obtained by applying Bayes' Theorem, and estimating:

$$p(blog|q) = \frac{p(q|blog) \cdot p(blog)}{p(q)}, \tag{1}$$

where $p(blog)$ is the probability of a blog and $p(q)$ is the probability of a query. Since $p(q)$ is a constant (for a given query), it can be ignored for the purpose of ranking. Thus, the probability of a blog being a key source given the query q is proportional to the probability of a query given the blog $p(q|blog)$, weighted by the *a priori* belief that a blog is a key source, $p(blog)$:

$$p(blog|q) \propto p(q|blog) \cdot p(blog). \tag{2}$$

Since we focus on a post-based approach to blog distillation, we assume the prior probability of a blog $p(blog)$ to be uniform. The distillation task then boils down to estimating $p(q|blog)$, the probability of a query q given a blog. For this estimation we consider a model based on language modeling techniques. We build a textual representation

of a blog, based on posts that belong the blog. From this representation we estimate the probability of the query topic given the blog's model. The language modeling setting allows us to use blog posts to build associations between queries and blogs in a transparent and principled manner.

4.1 A baseline model

Our baseline model for estimating the probability of a query given a blog, $p(q|blog)$, represents the blog by a multinomial probability distribution over the vocabulary of terms. Therefore, a blog model θ_{blog} is inferred for each blog, such that the probability of a term given the blog model is $p(t|\theta_{blog})$. The model is then used to predict how likely a blog would produce a query q. Each query term is assumed to be sampled identically and independently. Thus, the query likelihood is obtained by taking the product across all terms in the query:

$$p(q|\theta_{blog}) = \prod_{t \in q} p(t|\theta_{blog})^{n(t,q)}, \tag{3}$$

where $n(t, q)$ denotes the number of times term t is present in query q.

To ensure that there are no zero probabilities due to data sparseness, it is standard to employ smoothing. That is, we first obtain an empirical estimate of the probability of a term given a blog $p(t|blog)$, which is then smoothed with the background collection probabilities $p(t)$:

$$p(t|\theta_{blog}) = (1 - \lambda_{blog}) \cdot p(t|blog) + \lambda_{blog} \cdot p(t). \tag{4}$$

In Eq. 4, $p(t)$ is the probability of a term in the document repository. In this context, smoothing adds probability mass to the blog model according to how likely it is to be generated (i.e., published) by any blog.

To approximate $p(t|blog)$ we use the posts as a bridge to connect the term t and the blog in the following way:

$$p(t|blog) = \sum_{post \in blog} p(t|post, blog) \cdot p(post|blog), \tag{5}$$

We assume the post and blog to be conditionally independent, and therefore set $p(t|post, blog) = p(t|post)$. The probability $p(post|blog)$ expresses the importance of a given post within the blog. The simplest approach is to set this distribution to be uniform, i.e., all posts of a blog are equally important. This results in $p(post|blog) = posts(blog)^{-1}$, where $posts(blog)$ is the number of posts in the blog. In Section 5 we shall see an alternative way of setting this probability, based on blog specific features. Next, we discuss our estimation of the smoothing parameter λ_{blog}. It is followed by an experimental evaluation of our baseline model.

4.2 Smoothing parameters

It is well-known that smoothing can have a significant impact on the overall performance of language modeling-based retrieval methods [15]. For the smoothing parameter λ_{blog} in Eq. 4, we set λ_{blog} equal to $\frac{n(blog)}{\beta + n(blog)}$, where $n(blog)$ is the length of the blog (i.e., summarizing the length of all posts of the blog). Essentially, the amount of smoothing is proportional to the length of the blog (and is like Bayes smoothing with a Dirichlet prior [7]). So if there are very few posts in the blog then the model of the blog is more uncertain, leading to a greater reliance on the background probabilities. We set β to be the average blog length. That is $\beta = 170$ for the title-only index and $\beta = 17,400$ for the title-and-body index.

4.3 Assessing the baseline model

We report on the retrieval scores achieved by our baseline model; see Table 2 for the results.

Model	Fields	MAP	R-prec	P@5	P@10	MRR
Baseline	TB	.3272	.4023	.4844	.4844	.6892

Table 2. Results of our baseline retrieval model.

The scores obtained by our baseline model would have been ranked second (according to most measures) if submitted to the TREC 2007 Blog distillation task; cf. [6].

Ensuring our baseline model achieves state-of-the-art performance on the blog distillation task using posts as indexing units allows us to look for improvements over this baseline using blog specific features. The next section details on these features and how we implement these in our model.

5 Beyond the baseline

The document collection at hand, blog posts, has several specific features that could be used to improve blog retrieval effectiveness. We distinguish three types of feature: (i) document structure, (ii) temporal structure, and (iii) social structure. Below we discuss each of these types and show how they can be implemented in our blog distillation model. Results obtained using the various implementations are discussed in Section 6.

5.1 Document structure

There are (at least) two blog specific document structure characteristics that can be incorporated in the retrieval model: (i) document length and (ii) representation. The former is implemented using a prior probability of a post being relevant, the latter is achieved using a linear combination of two content representations.

Blog posts are characterized by relatively short document lengths. For a blog post to be considered relevant to a given topic though, it should contain "enough" information. Short posts, containing mostly uninformative utterances of a blogger, have a smaller a priori probability of being relevant than longer posts. We model this feature as the prior probability of a post $p(post)$. We rewrite the $p(post|blog)$ component from Eq. 5 using Bayes' Theorem:

$$p(post|blog) = \frac{p(blog|post) \cdot p(post)}{p(blog)}, \tag{6}$$

where $p(blog)$ is constant for all posts in a blog, and $p(blog|post)$ is set to 1. We therefore set $p(post|blog) \propto p(post)$. In the case of post length we set $p(post) \propto log(|post|)$, where $|post|$ is the length of the post in words.

The second document structure characteristic we use is title (T) versus title+body (TB) representation. A blog post consists of a title and a body part. Since the title usually is a very clean, yet short description of the post content, we expect a precision-enhancing effect from using the T index instead of the TB index. On the other hand, due to the very short documents, we believe a T only run cannot achieve state-of-the-art performance on recall-based metrics (e.g., MAP). We therefore explore the possibilities of a mixture of two representations: The rationale behind mixing two content representations is to mimic a user's search behavior: after being presented with a relevant blog post, a user might look at the titles of other posts

within the same blog to come to a final relevance judgement concerning the entire blog. We mimic this behavior by combining the T and TB representations in a linear way:

$$p(q|\theta_{blog}) = \lambda_{TB} \cdot p_{TB}(q|\theta_{blog}) + (1 - \lambda_{TB}) \cdot p_T(q|\theta_{blog}), \quad (7)$$

where both $p_{TB}(q|\theta_{blog})$ and $p_T(q|\theta_{blog})$ are defined in Eq. 3. In the final combination of this linear combination run with the other features, we apply the non-uniform probabilities $p(post|blog)$ to the TB run only, and keep this probability uniform in the T run (i.e. $p(post|blog) = posts(blog)^{-1}$).

5.2 Temporal structure

Blogs have a very specific temporal structure in that posts within a blog are time-stamped and reverse chronologically ordered (most recent post first). When searching for blogs devoted to a given topic, one could assume the most recent posts to be of more importance than much older posts: even a blogger's interest can shift over time, and more recent posts can therefore give better insight in their current interests. To incorporate this intuition in our model we assign a recency score (rs) to each post in a blog:

$$rs(post, blog) = \begin{cases} 1 + \gamma, & \text{if } recency(post, blog) \leq M \\ 1, & \text{otherwise,} \end{cases} \quad (8)$$

where posts within the top M most recent posts within a blog are awarded with γ additional points, on top of the standard 1 for each post. We normalize the recency scores to get an estimate of $p(post|blog)$:

$$p(post|blog) = \frac{rs(post, blog)}{\sum_{post' \in blog} rs(post', blog)}. \quad (9)$$

5.3 Social structure

Probably the most eye-catching feature of blogs is their social structure: this structure displays itself in more than one way (e.g., blog rolls), but we focus on the commenting feature of blogs. Readers of a post can usually respond to that post by leaving a comment; the more comments readers leave with a certain post, the more important we consider this post to be. As with the document length, we implement this feature using a prior probability $p(post)$ (following Eq. 6). We set $p(post) \propto 1 + \log(comments(post))$, where $comments(post)$ is the number of comments for a post.

6 Results

In this section we present the outcomes of our experiments, followed by the main observations from these outcomes and a more detailed analysis of these outcomes.

All results are summarized in Table 3; significance is tested using a two-tailed paired t-test, with significant differences compared to the baseline TB run reported with ** for $\alpha = .01$, and * for $\alpha = .05$. For runs (A) and (F) we need an estimation of λ_{TB}, the weight of the title+body index (see Eq. 7). We estimate this parameter empirically by performing a sweep over possible λ_{TB} values. We obtained best performance using $\lambda_{TB} = .7$ for run (A) and $\lambda_{TB} = .8$ for run (F).

Run	MAP	R-prec	P@5	P@10	MRR
Baseline					
title+body	.3272	.4023	.4844	.4844	.6892
title	.2602**	.3236**	**.5689***	.4889	**.7770**
Document structure					
(A) T+TB	.3449	.4186	.5200	**.5044**	.7733**
(B) doc. length	.3464*	.4196	.5111	.4733	.6820
Temporal structure					
(C) recency	.3323*	.4031	.4844	.4822	.6866
Social structure					
(D) comments	.3278	.4066	.4978	.4822	.6798
Combinations					
(E) B+C+D	.3452**	.4148*	.5156	.4822	.7245
(F) A + E	**.3596****	**.4277***	.5467*	**.5044**	.7654**

Table 3. Results of the baseline runs, single feature runs, and combined runs, using MAP, R-precision, P@5, P@10 and mean reciprocal rank.

6.1 Observations

We formulate the main observations based on our experiments. First, our baseline model displays state-of-the-art performance using a blog post index and no special features (Table 2). Further main observations include the influence of the title index, the blog specific features, and the combination of all previously presented components. Finally, we pick three example topics that are discussed in more detail.

Title-only index: We see that by using a title only index we can improve on early precision (P@5, P@10) and mean reciprocal rank, where the difference on P@5 is significant. This gain in precision has the side-effect of a loss in recall, as shown by significant drops in MAP and R-precision scores. Still, this title only run would have been ranked third in TREC 2007 (ordered by MAP).
When we combine the T and TB runs we improve over the baseline on all metrics. This result does show that using the leaner content representation of the title only index helps improving blog distillation performance when combined with the full post index.

Blog specific features: As we can see from Table 3 all three blog features (post length, recency, and comments) help to improve over the baseline on most metrics. Especially post length seems a good indicator of post importance, whereas the number of comments shows only marginal improvements. When we combine all three features by averaging over their probabilities, we see an increase over the baseline on all metrics except P@10. From the fact that improvements of the combined run on MAP and R-precision are more significant than their single run counterparts, we can conclude that the combination has a recall-enhancing effect.

Combination of features: The combination of the TB run with all features and the T run shows best overall performance. It achieves highest scores on 3 of 5 metrics and improves significantly over the baseline on MAP, R-precision, P@10, and MRR. The performance in terms of MAP is close to the best TREC 2007 run (MAP .3695).

6.2 Topic analysis

Zooming in on the results of the final run (F) compared to the baseline, we observe AP gains for most topics (see Figure 1: 32 out of 45 topics gain over the baseline, while 13 drop). Also, the improvements in AP are in general stronger than the AP drops. We select the two topics with the largest improvement (990 and 982), and the topic with the biggest AP drop (967):

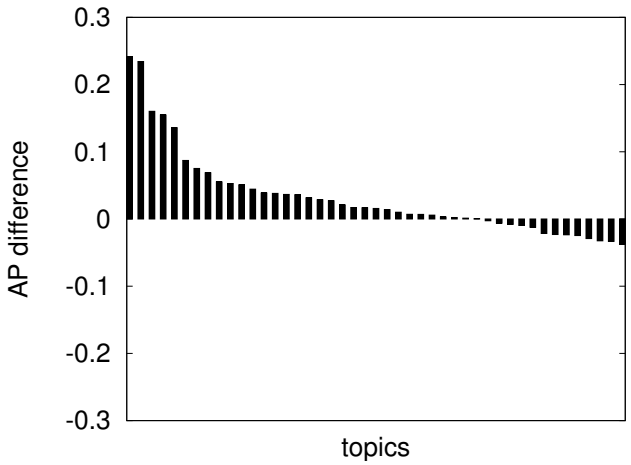

Figure 1. AP difference for run (F) compared to baseline.

Topic 990 (lost tv): has an AP of .1913 in the baseline setting, but shows a remarkably high performance in the title only run of AP .5535. The final score for this topic is .4326, showing that the title only run can outperform the combination.

Topic 982 (machine learning): has a baseline AP of .5657; using the post importance probabilities we can improve this to .6143, but most improvement is obtained by using the title only run. This run shows an AP performance of .8000, which is also the final score for this topic.

Topic 967 (home baking): has a baseline AP of .2703; using only the number of comments as prior probability, we can improve to .2895. On the other hand, all other features, but especially the title only run (.0884) and length prior (.1783) hurt performance. The final score (.2329) reflects this, although the decrease is only marginal.

7 Conclusion

In this paper we described a language modeling approach to the task of identifying blogs that are principally devoted to a given topic. Our main focus concerned an approach that uses blog posts as indexing units, instead of the (aggregated posts of) blogs that have so far mostly been used. There are a number of pragmatic reasons that account for this choice: (i) to allow for easy incremental indexing, (ii) for presentation of retrieval results posts are natural and coherent units, and (iii) to allow the use of one index for both blog post and blog retrieval. At the same time we aimed to achieve state-of-the-art performance. On top of this, we wanted to deploy blog specific features to improve over our baseline model.

Our main finding is that the we can indeed achieve state-of-the-art performance using a blog post index. Additionally, we find that (i) the lean title-only content representation has a clear precision-enhancing effect when compared to a title+body representation, while a combination of the two representations outperforms both; (ii) post priors based on post length and number of comments have a positive influence on retrieval performance; (iii) temporal ordering of posts can be used as an indication of importance of a post given a blog; and (iv) a combination of all blog specific features on top of our baseline shows significant improvements over the baseline on most metrics.

As to future work, we are interested in combining our proposed post-based approach to blog distillation with techniques that have been shown to improve retrieval effectiveness for the task at hand, including query enrichment, link analysis, credibility scoring [14], and spam filtering.

Acknowledgements

We would like to thank our reviewers for their valuable feedback. Weerkamp and De Rijke were supported by the E.U. IST programme of the 6th FP for RTD under project MultiMATCH contract IST-033104. Balog and De Rijke were supported by the Netherlands Organisation for Scientic Research (NWO) under project number 220-80-001. De Rijke was also supported by NWO under numbers 017.001.190, 640.001.501, 640.002.501, STE-07-012.

REFERENCES

[1] J. Elsas, J. Arguello, J. Callan, and J. Carbonell. Retrieval and feedback models for blog distillation. In *TREC 2007 Working Notes*, 2007.

[2] K. Fujimura, T. Inoue, and M. Sugisaki. The eigenrumor algorithm for ranking blogs. In *WWW 2005 Proceedings*, 2005.

[3] K. Fujimura, H. Toda, T. Inoue, N. Hiroshima, R. Kataoka, and M. Sugizaki. Blogranger—a multi-faceted blog search engine. In *Proceedings of the WWW 2006 3nd Annual Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics*, 2006.

[4] W.-L. Lee and A. Lommatzsch. Feed distillation using adaboost and topic maps. In *TREC 2007 Working Notes*, 2007.

[5] C. Macdonald and I. Ounis. The TREC Blogs06 collection: Creating and analyzing a blog test collection. Technical Report TR-2006-224, Department of Computer Science, University of Glasgow, 2006.

[6] C. Macdonald, I. Ounis, and I. Soboroff. Overview of the TREC 2007 Blog Track. In *TREC 2007 Working Notes*, pages 31–43, 2007.

[7] D. J. C. Mackay and L. Peto. A hierarchical dirichlet language model. *Natural Language Engineering*, 1(3):1–19, 1994.

[8] G. Mishne. *Applied Text Analytics for Blogs.* PhD thesis, University of Amsterdam, 2007.

[9] G. Mishne and M. de Rijke. A study of blog search. In M. Lalmas, A. MacFarlane, S. Rüger, A. Tombros, T. Tsikrika, and A. Yavlinsky, editors, *Advances in Information Retrieval: Proceedings 28th European Conference on IR Research (ECIR 2006)*, volume 3936 of *LNCS*, pages 289–301. Springer, April 2006.

[10] I. Ounis, C. Macdonald, M. de Rijke, G. Mishne, and I. Soboroff. Overview of the TREC 2006 Blog Track. In *The Fifteenth Text Retrieval Conference (TREC 2006)*. NIST, 2007.

[11] S. Robertson and J. Callan. Routing and filtering. In *TREC Experiment and Evaluation in Information Retrieval*, pages 99–122. MIT, 2005.

[12] K. Seki, Y. Kino, and S. Sato. TREC 2007 Blog Track Experiments at Kobe University. In *TREC 2007 Working Notes*, 2007.

[13] J. Seo and W. B. Croft. UMass at TREC 2007 Blog Distillation Task. In *TREC 2007 Working Notes*, 2007.

[14] W. Weerkamp and M. de Rijke. Credibility improves topical blog post retrieval. In *ACL08:HLT*, June 2008.

[15] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Trans. Inf. Syst.*, 22(2):179–214, 2004.