FEUP at TREC 2010 Blog Track: Using h-index for blog ranking

José Luís Devezas[†], Sérgio Nunes[‡], Cristina Ribeiro[‡]
[†] Labs SAPO/UP
[‡] DEI, Faculdade de Engenharia, Universidade do Porto
Rua Dr. Roberto Frias, s/n 4200-465 Porto, Portugal
joseluisdevezas@gmail.com, {ssn,mcr}@fe.up.pt

Abstract

This paper describes the participation of FEUP, from the University of Porto, in the TREC 2010 Blog Track. FEUP participated in the baseline blog distillation task with work focused on the use of link features available in the TREC Blogs08 collection. The approach presented in this paper uses the link information available in most individual posts to amplify each post's score. Blog scores, and subsequent ranks, are obtained by combining individual post scores. We boost post scores using the in-degree of each post and the h-index of each blog. This results in an improvement of P@10, over our baseline, for the in-degree and the h-index runs. When compared to the in-degree, the h-index run results in higher performance values for each of the applied evaluation metrics.

1 Introduction

In this paper we describe the participation of a group from the Faculty of Engineering of the University of Porto (FEUP) in the TREC 2010 Blog Track. FEUP's participation was focused on the exploration of link evidence for the baseline blog distillation task. In this year's edition of the TREC Blog Track, we consider the link features present on post bodies in the form of HTML anchors. Based on the extracted URLs, we build the post graph for the Blogs08 collection, on top of which we base our work.

We extend preliminary work on this topic by Branco [1], exploring and evaluating his approach to blog ranking, based on the h-index [2], when combined with other ranking functions. We compare the h-index (a link-based metric), with other link-based metrics, like the in-degree, studying their influence on the retrieval process.

2 Blogs08 Collection Overview

The Blogs08 test collection was released in early April 2009 and became the official collection for the TREC Blog Track in the 2009 edition. To prepare this collection, a total of 1,303,520 feeds were polled once a week from January 14th, 2008 to February 10th, 2009 (394 days). The polled feeds, associated permalinks and homepage documents were stored, resulting in a collection with a total compressed size of 453 GB. Table 1 presents an overview of the Blogs08 collection, including statistics about link usage. Overall, the collection has over 3.4 billion links, including multiplicity and self-citations. Nearly every document (98.9%) has at least one link. Documents/posts with links always have at least one link pointing to feeds/blogs in the collection. Even though only 10.9% of the links point to blogs in the collection, this fraction represents over 374 million links for more than 28 million posts — resulting in 13 links per post.

Blogs08 Collection		
Total no. of feeds	1,303,520	
Total no. of documents	28,488,766	100.0%
| with links to the web	28,162,094	98.9%
| with links to the collection	28,162,094	98.9%
Total no. of links	3,434,507,661	100.0%
| to the collection	374,598,062	10.9%
Avg. no. of links per doc.	121	
| to the collection	13	
Avg. no. of links per feed	2,635	
| to the collection	570	

Table 1: Blogs08 collection characteristics.

3 System Overview

The Terrier information retrieval platform [3] was used to index the collection at the document level, excluding the tags `DOCHDR`, `DATE_XML`, `FEEDNO`, `BLOGHPNO`, `BLOGHPURL` and `PERMALINK`. For query expansion, we use the `Bo1` term weighting model and, for the retrieval task, we use Terrier's implementation of `BM25` [4], with the default parameters $k_1 = 1.2$, $k_3 = 8$ and $b = 0.75$. We also take advantage of the document prior features, introduced in Terrier 3.0, by using a `SimpleStaticScoreModifier` that applies a query-independent score, with a given weight w, to each document. The weighted feature $(w \times prior(d))$ is then added to the document's query-dependent score. For document prior features, we experiment with the in-degree of the document and with the h-index of the document's feed, both calculated based on the post graph.

The indexing and retrieval tasks are carried in a server equipped with an Intel® Core$^{\mathrm{TM}}$ 2 Quad CPU Q9300 @ 2.50GHz and 8GB of RAM. There is a limit of 2GB of memory we can address for the heap of the Java Virtual Machine, given the 32 bit architecture. The graph mining process is carried in a different machine, equipped with an Intel® Xeon® CPU X5450 @ 3.00GHz and 32GB of RAM.

4 Graph Mining

The link features we select are based on the explicit connections between blog posts (or documents). We define an explicit connection as any value of the `href` attribute, found in HTML anchors, that starts with "`http://`" or "`https://`". We parse every document, extracting the links that fit this criteria, and build a file where we associate the extracted URLs to the corresponding `DOCNO`. We parse this file, converting each URL belonging to Blogs08 collection to its respective `DOCNO` and discarding any URL that points to resources outside the collection. This results in a post graph for the Blogs08 collection, in the format $\texttt{DOCNO}_{source}\rightarrow\{\texttt{DOCNO}_{target}\}$. We also create an inverted representation of the graph, which is used to compute the in-degree of the posts and the h-index of the blogs.

The definition of h-index is presented next, together with an explanation of its application in the context of blogs. An author has an index h if h of their publications has at least h citations, and the rest has less than h citations. The h-index takes into account both the number of citations and the sustainability. This means that if an author is the holder of a highly cited article, the h-index will still be bounded by the number of publications. Similarly, if an author has published a great number of articles, the h-index will be limited by the number of citations. Figure 1 establishes the analogy between a publication network and a blog network. Each blog takes the role of an author, with hyperlinks between posts being compared to article citations. So, for example, a blog has h-index 5 if 5 of its posts have 5 or more in-links and the rest has less than 5 in-links. This metric seems ideal to boost active blogs — that publish posts frequently — with a high number of in-links.

We did not remove edge multiplicity or loops from the post graph, however we believe this approach should be explored in the future. Since the h-index is used as a bibliometrics measure, a citation to article x found in article y is only counted once, independently of the number of times the author references article x within article y. On the other hand, an author never cites an article within itself, so loops are something

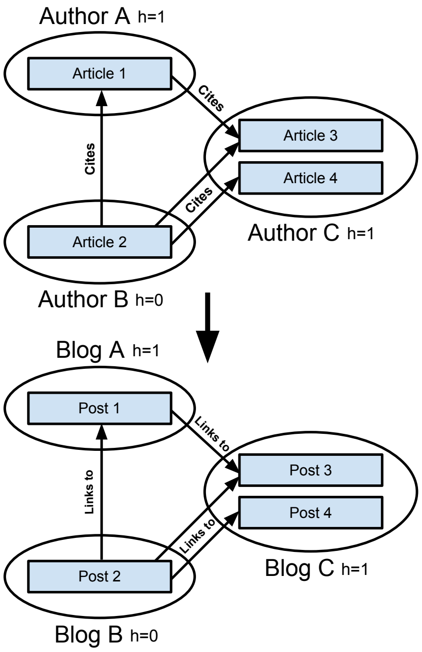

Figure 1: Abstraction used for the application in the h-index to the blog context.

we should also discard from the post graph in future explorations, even more so given the analogy to bibliometrics and self-citations aren't a direct sign of popularity or relevance.

5 Blog Distillation

We participate in the baseline blog distillation task, ignoring any facets attached to the topics, therefore submitting only two runs. For each run, we use the merged 2009 and 2010 topics, making a total of 100 topics.

5.1 Baseline

In order to study the impact of link-based ranking on the blog distillation process, we define a baseline based on Terrier's implementation of BM25 weighing model. Retrieval is made at the document level. Feed scores are then calculated by adding the post scores of each feed and then dividing by the total number of posts in the corresponding feed. We used the topics and relevance assessments from 2009 to optimize the P@10 and R-prec retrieval performance metrics for the in-degree and h-index based ranking functions when combined with this baseline.

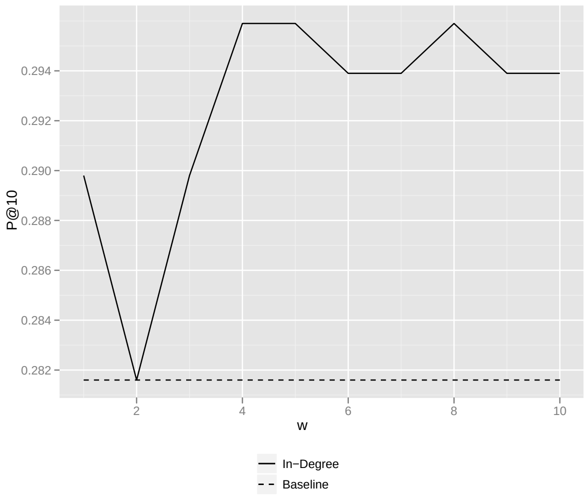

Figure 2: Evolution of the average P@10 for values of $w \in \{1..10\}$, considering all 50 topics for TREC 2009 Blog Track.

The link-based metrics were introduce at the document level, prior to calculating the feed score. As query-independent features, these metrics were included in the final score as priors, simply by adding to the already calculated BM25 document score. The h-index value is not directly associated with a document, but instead with the feed it belongs to, so we combine the h-index value of the corresponding feed with the document score.

Since we can only submit two runs for the baseline blog distillation task and our focus is on studying the effectiveness of the h-index when compared to other link-based weighting measures, we submitted two baseline runs, one using the BM25 combined with the in-degree and the other using the BM25 combined with the h-index.

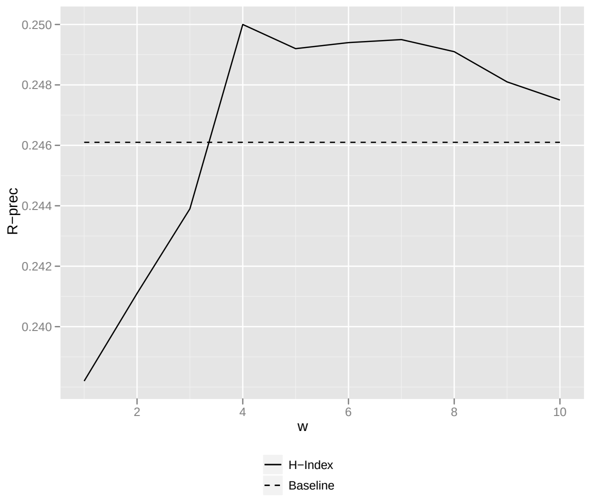

Figure 3: Evolution of the average R-prec for values of $w \in \{1..10\}$, considering all 50 topics for TREC 2009 Blog Track.

5.2 Boost Feed Score by In-Degree

In the first run, tagged `FEUPirlab1`, we boost the BM25 score for each document by combining it with the in-degree of the document, directly acquired from the previously built post graph. So, for a query q, the score for document d is given by $score(d|q) = BM25(d|q) + w \times log(in(d))$, where $in(d)$ is the in-degree for a post or document d. The optimization of w is done by preparing a run with the 50 topics from 2009, which is then evaluated by calculating the retrieval performance for values of w between 1 and 10, using unitary steps.

We calculate the values for several performance metrics — b-Pref, R-prec, P@10 and MAP — and select the value of w that results in an obvious maximum for one of the metrics. Based on P@10 (Figure 2), we use $w = 4$ to calculate the in-degree score component of each document.

5.3 Boost Feed Score by H-Index

In the second run, tagged `FEUPirlab2`, we boost the BM25 score for each document by combining it with the h-index of the document's feed. We calculate the h-index from the in-degree frequency table, using the analogy to bibliometrics, previously illustrated in Figure 1.

So, for a query q, a document d's score is given by $score(d|q) = BM25(d|q) + w \times log(h[feed(d)])$, where $h[feed(d)]$ is the h-index of the document's feed. The w constant is optimized as described in Section 5.2. Based on R-prec (Figure 3), we use $w = 4$ to calculate the h-index score component of each document.

6 Results

Table 2 presents an overview of the results for this year's baseline and faceted blog distillation task. As expected from our experiments, there is a slight, consistent improvement — of approximately 1% — when the h-index is used over the in-degree. When compared to the BM25 baseline, we achieve higher performance values for R-prec and P@10. This clearly indicates that the introduction of link-based metrics allows for an improvement of the relevance for the top n results. When we account for the order and look at MAP and b-Pref values, we verify that this improvement isn't noticeable. The h-index run has better results than the in-degree, however they are not statistically significant.

7 Related Work

Branco has already explored the h-index as a link analysis metric for blog ranking. We conduct here a more comprehensive and controlled assessment of this approach. Based on this idea, we combined the h-index with query-dependent ranking functions, applying it to a much larger collection (26 times more blogs and 10 times more posts), aiming at finding an optimal weight for the h-index component while examining the gain introduced in the retrieval system.

8 Conclusions

TREC Blogs08 collection comprises a large number of blogs, with posts containing a large number of link

Run	Description	MAP	b-Pref	R-prec	P@10
BM25	Local baseline (not submitted).	0.1942	0.2108	0.2446	0.2849
stdbaseline1	First TREC's selected baseline.	0.2174	0.1921	0.2427	0.1875
stdbaseline2	Second TREC's selected baseline.	0.1720	0.1334	0.1726	0.1542
stdbaseline3	Third TREC's selected baseline.	0.1461	0.1230	0.1645	0.1667
FEUPirlab1	BM25 combined with the in-degree.	0.1830	0.1998	0.2425	0.3068
FEUPirlab2	BM25 combined with the h-index.	0.1911	0.2077	0.2570	0.3178

Table 2: Results of the baseline blog distillation subtask.

references. Given our interest in studying the influence of link analysis scores in the retrieval process, having access to a collection with rich link evidence was fundamental.

Experimenting with the h-index for blog ranking in a query-dependent scenario has shown a positive effect in improving the number of relevant blogs for the top results. However, in our experiments, for performance metrics that take into consideration the order of the results, we were not able to improve on the baseline values.

We intend to explore other link-based ranking functions, like PageRank, establishing a comparison between its performance and that of h-index, for TREC Blogs08 collection.

We would also like to study how edge multiplicity and loops influence h-index ranking in the blogosphere, while exploring different values for w and attempting to achieve more consistent results for the various performance metrics.

9 Acknowledgements

We would like to thank the laboratory colleagues for the input on IR performance measures and constant availability to discuss ongoing issues. This work was partially supported by a research grant from Labs SAPO/UP.

References

[1] J. Branco. *Aplicação do h-index em blogues*. Master's thesis, Faculty of Engineering, University of Porto, 2008.

[2] J. Hirsch. An index to quantify an individual's scientific research output. *Proceedings of the National Academy of Sciences of the USA*, 102(46):16569–16572, 2005.

[3] I. Ounis, G. Amati, V. Plachouras, B. He, and C. Macdonald. Terrier: A high performance and scalable information retrieval platform. *Proceedings of ACM SIGIR'06 Workshop on Open Source Information Retrieval (OSIR 2006)*, 2006.

[4] S. E. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford. Okapi at TREC-3. In *Proceedings of the 3rd Text REtrieval Conference TREC3*, pages 109–126. NTIS, 1995.