NovaSearch at TREC 2013 Federated Web Search Track: Experiments with rank fusion

André Mourão, Flávio Martins and João Magalhães

Faculdade de Ciências e Tecnologia
Universidade Nova de Lisboa
Caparica, Portugal
a.mourao@campus.fct.unl.pt, flaviomartins@acm.org,
jm.magalhaes@fct.unl.pt

Abstract. We propose an unsupervised late-fusion approach for the results merging task, based on combining the ranks from all the search engines. Our idea is based on the known pressure for Web search engines to put the most relevant documents at the very top of their ranks and the intuition that relevance of a document should increase as it appears on more search engines [9].
We performed experiments with state-of-the-art rank fusion algorithms: RRF and Condorcet Fuse and our proposed method: Inverse Square Rank (ISR) fusion algorithm. Rank fusion algorithms have low computational complexity and do not need engines to return document scores nor training data. Inverse Square Rank is a novel fully unsupervised rank fusion algorithm based on quadratic decay and on logarithmic document frequency normalization. The results achieved in the competition were very positive and we were able to improve them further post-TREC.

1 Introduction

Federated search techniques search on multiple search engines simultaneously. The Federated Web Search (FedWeb) track as designed "to evaluate approaches to federated search at very large scale in a realistic setting, by combining the search results of existing web search engines." [1]

Web search engines may target different categories (e.g. news, blogs, articles) and retrieve multiple data types (e.g. web pages, video, images). Some federated retrieval systems try to guess the query category and tailor the final rank to emphasize results from the engines of the category detected. However, regardless of data type or category, all search engines share the basic idea of analysing and indexing documents to produce relevant ranks for user queries. Our main motivation is to leverage the knowledge of a number of web search engines through the combination of their ranks.

The 2013 track was focused on both resource selection (Task 1) and results merging (Task 2). Since our approach was based on unsupervised fusion algorithms using all search engines, we did not require resource selection. Thus, we only participated on the results merging task.

[1] https://sites.google.com/site/trecfedweb/

The design of the tasks allows a multitude of approaches and techniques: (Use full documents vs. only snippets; external data vs. no external data; supervised vs. unsupervised). Our participation is an unsupervised technique for combining the ranks of all engines in a late fusion approach.

Rank fusion aims at combining ranked document lists (ranks) from multiple sources into a single (combined) ranked list. Unsupervised methods can be divided into score-based fusion (CombSUM and variants [8]), rank-based fusion (RR [10] and RRF [2]) and voting algorithms [5, 1]. For the specific case of federated Web search, score-based fusion is not feasible, as search engines do not provide a score for the documents, therefore we focused on rank and voting algorithms.

This paper is organized as follows: section 2 details our merging approach, section 3 contains the evaluation and section 4 contains the conclusion.

2 Rank fusion: Standing on the shoulders of giants

Vogt et Cottrell [9] observe the two effects related to search engine fusion:

- The Skimming effect: search engines are designed to put relevant documents at the very top;
- The Chorus effect: documents that appear on multiple ranks are more relevant.

The Skimming effect is present in the inherent motivation of web search engines to rank relevant documents at top positions. Thus, the higher the rank of a document, the most important it is for the query.

The Chorus effect is present in the idea that, if a document is deemed relevant by multiple engines, the probability of it being relevant increases. For our experiments, we simply assume that different documents are identified by URL, meaning that documents with the same URL are duplicates.

Table 1. Document frequency across search engines.

Doc. freq.	Doc. count	% of doc.
1	130058	97.26%
2	2258	1.69%
3	755	0.56%
4	330	0.25%
5	213	0.16%
6	79	0.06%
7	24	0.02%
8	7	0.01%

Table 1 contains document frequency values across search engines on the 2013 Federated Web Search dataset. The table shows that about 97% of documents

appear only in one engine. In line with the Chorus effect, we believe that the the remaining 3% should be ranked higher on the final rank.

To support our hypothesis, we performed a post-TREC analysis of the dataset of document relevance versus frequency. For each query, we analysed the relevance of the documents for all engines and measured the document frequency across engines. Figure 1 contains the average results for all queries. The document frequency axis represents the number of search engines where a certain document appears and the Percentage of documents axis represents the percentage of documents divided by relevance level. The increase in relevance with frequency is clear: 12% of the documents that appear only one engine are relevant vs. over 75% of relevant results for documents with a frequency of 4 or more. The plot also shows that the increase in relevance occurs in all relevance levels, meaning that documents that appear on more engines are also more likely to have an higher relevance level.

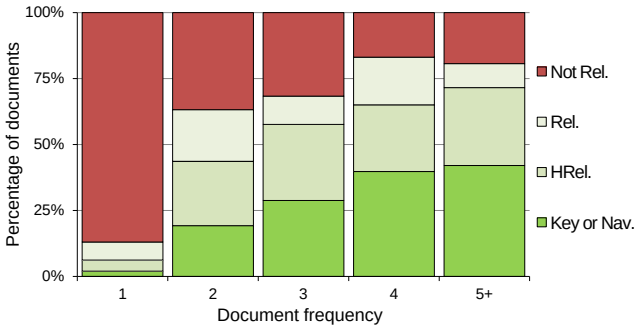

Fig. 1. Document frequency across engines versus relevance for all queries. Not Rel.: not relevant; Rel.: relevant; HRel.: Highly relevant; Key or Nav.: Key or navigational document

3 Rank fusion techniques for result merging

Inspired by the described effects, our idea was to apply rank fusion techniques for merging the results from all the engines. Unsupervised rank fusion techniques do not need to give weighs to search engines; they combine the results from all provided search engines without the need for resource selection.

The initial step of the process is to transform the provided result snippets into ranks. Consider the following sorted rank for the search engine E for the query q:

$$\mathrm{E}(q) = \{(d_1, 1), ..., (d_k, k), ..., (d_i, i)\} \quad (1)$$

d_1 represents the most relevant document (ranked 1), d_k represents document ranked k and i represents the size of a rank (number of documents retrieved by the engine). For the given dataset, i was limited to a maximum value of 10.

In the provided snippets (Figure 2), documents are identified with a local unique id (LID) tied to the engine and query (e.g. FW13-e176-7027-01 is the rank 1 document for query 7027 in engine 176).

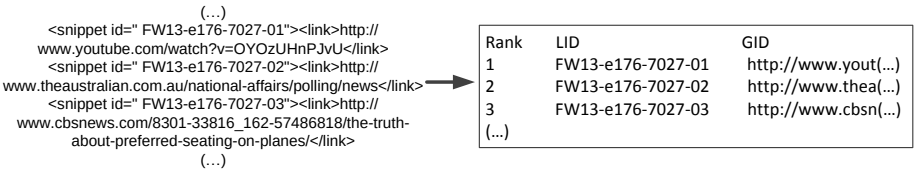

Fig. 2. Conversion from snippets to a rank

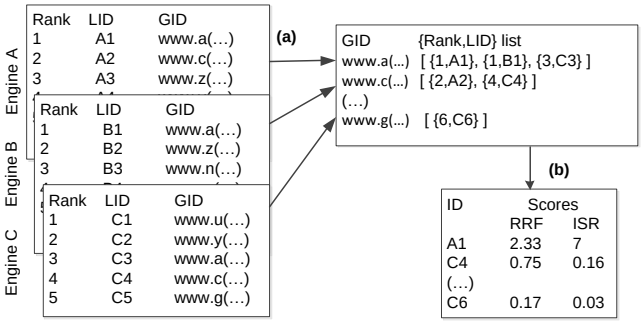

Fig. 3. Rank-based fusion example. Initially, the documents in the ranked lists are grouped by GID (a). Individual document score is computed from the Rank,LID lists and the new combined rank is sorted by new scores (b).

To find similar documents across ranks, Figure 3 (a), it is necessary to use the documents URL as a global unique document identifier (GID). GID are converted into one of the corresponding LID after fusion, Figure 3 (b), to conform with the submission format.

This approach to find duplicated documents is fast and only requires a global identifier; it is completely agnostic to document content or category. This means that is can be deployed instantly on existing systems from a variety of domains with minimal performance requirements. It performs well in most cases, only failing on edge cases (e.g. engines retrieving the same 404 page for multiple queries, engines that retrieve different documents with the same URL).

Having formalized rank representation and duplicate detection, the final re-ranking score must be computed. The new ranks are sorted descending by the re-ranking score.

The re-ranking is computed using rank fusion methods. Rank fusion methods take the ranked results and calculate a new score using the document ranks and frequency across multiple results lists, Figure 3 (b). Existing methods are Reciprocal Rank [10] (RR) and Reciprocal rank fusion [2]) (RRF). RRF can be defined as:

$$\mathrm{RRF}(d) = \sum_{e=1}^{N(d)} \frac{1}{k + R(e,d))} \tag{2}$$

A typical value for k (and the one we used on our experiments) is 60.

The motivation behind our method ISR, is that existing techniques underestimate the weight of document frequency and overestimate the importance of low-rank documents. ISR can be defined as:

$$\mathrm{ISR}(d) = N(d) \times \sum_{e=1}^{N(d)} \frac{1}{R(e,d)^2}. \tag{3}$$

Where *N(d)* is the number of times a document appears on a results list (document frequency), and *R(e,d)* is the rank of document d on engine e.

ISR uses document frequency as an explicit multiplier (*N(i)*) and proposes a faster document score decay as rank increases ($R(e,d)^2$ vs. $R(e,d)$)

The simplest form of the ISR technique weights document frequency using the absolute document frequency. We observed that linear weighting over-emphasises documents present on multiple ranks and fails to penalize documents that appear in a single rank. In our initial experiments, penalizing documents that appear on a single rank, combined with logarithmic document frequency weighting, leads to a significant performance improvement. Inspired by BM25L [4] (where the logarithm was introduced to counteract increased score on long documents), we identified logarithmic normalization to provide a good model for document frequency weighting. The other functions in the ISR family are log_ISR and normalized log_ISR (logN_ISR).

$$\mathrm{log\_ISR}(d) = log(N(d)) \times \sum_{e=1}^{N(d)} \frac{1}{R(e,d)^2}. \tag{4}$$

$$\mathrm{logN\_ISR}(d) = log(N(d) + \sigma) \times \sum_{e=1}^{N(d)} \frac{1}{R(e,d)^2}. \tag{5}$$

We set $\sigma = 0.01$ on this paper, based on previous experiments.

For TrecFedWeb 2013, we tested ISR [6], along with existing rank-based fusion (RRF [2]) and a voting approach, Condorcet Fuse [5].

4 Evaluation

4.1 Dataset

The 2013 TREC FedWeb dataset is an extension to the 2012 TREC FedWeb dataset [7]. It contains a collection of search results sampled from 157 search engines over 200 queries. Each search engine is related to one or more search categories, such as web, news, travel, and video. Relevance judgments span 5 categories: Navigational (rel = 1), Key (rel = 1), Highly relevant (rel = 0.5), relevant (rel = 0.25) and Not relevant (rel = 0).

4.2 Experiments

We submitted runs using the described approach for the result merging task. The differences between the runs was the fusion method: ISR (nsISR), RRF [2] (nsRRF) and CondorFuse [5] (nsCondor). All runs use all search engines equally for fusion, no resource selection or weighting was performed. The results are in Table 2:

Table 2. 2013 TREC FedWeb 2013 results merging results for all queries.

Runs	Algorithm	nDCG	nDCG@20	nDCG@50	nDCG@100	P@10
nsRRF	RRF	0.5082	0.2569	0.2347	0.2553	**0.3700**
nsISR	ISR	0.4793	0.1654	0.1635	0.1990	0.3100
nsCondor	Condor	0.4690	0.1353	0.1550	0.1993	0.2780
Post-mortem results						
-	log_ISR	**0.5229**	**0.2620**	**0.2763**	**0.2904**	0.3660
-	logN_ISR	0.5122	0.2613	0.2393	0.2594	0.3680

All the runs obtained good global results, and our RRF run achieved the best global nDCG@20 [3], which as the primary evaluation metric. The other runs were also above the the median for some metrics.

In addition to the submitted runs, we performed further experiments with the provided relevance judgements. We tested the logarithmic variants of ISR: Log_ISR and LogN_ISR. These variants focus on logarithmic weighting for document frequency (contrasting with the linear weighting for ISR).

The results are very positive, beating our previous results by a significant margin and even beating maximum nDCG@20 and nDCG@50.

5 Conclusions and discussion

In this paper, we describe our participation at TREC FedWeb 2013. We propose a unsupervised approach for the results merging task, based on combining the

ranks from all the search engines (late fusion). Our approach is completely unsupervised and does not require any external data; we only used the provided result snippets. We tested existing rank fusion methods and our novel proposal, ISR.

According to the provided global results for TREC FedWeb 2013, our run with the RRF fusion algorithm (nsRRF), was the best performing approach in terms of nDCG@20 and got results above the median for other tested metrics [3]. In our post-TREC experiments, ISR variants were able to improve results even further on all metrics, keeping the good performance and lack of need for supervision inherent to these methods.

The detection of duplicated documents greatly influenced the results. Our technique (URL matching) was designed to be fast and analyse only the snippets (vs. analysing document content), and missed some similar or near-similar pages. The organizers [3] detected duplicates using a more thorough set of techniques that account for document similarity (MD5 hash and Simhash) and an exclusion list of pages that contain false positives URL duplicates. A similar false positive exclusion list can be included in future iterations of the algorithm.

We believe rank fusion is an important area in information retrieval and federated search can greatly benefit to novel developments in these area. Rank fusion approaches are unsupervised, do not require external data and are computationally inexpensive, enabling real time use. ISR approaches allow improved retrieval performance with low complexity, by harnessing the power of multiple individual search engines.

Acknowledgments

This work has been partially funded by the Portuguese National Foundation under the projects PTDC/EIA-EIA/111518/2009 and UTA-Est/MAI/0010/2009.

References

1. Aslam, J.A., Montague, M.: Models for metasearch. In: Proceedings of the 24th SIGIR. SIGIR '01 (2001)
2. Cormack, G.V., Clarke, C.L.A., Buettcher, S.: Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In: SIGIR '09 (2009)
3. Demeester, T., Trieschnigg, D., Nguyen, D., Hiemstra, D.: Overview of the trec 2013 federated web search track. In: TREC (2013)
4. Lv, Y., Zhai, C.: When documents are very long, bm25 fails! In: Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval. pp. 1103–1104. ACM (2011)
5. Montague, M., Aslam, J.A.: Condorcet fusion for improved retrieval. In: Proceedings of the 11th ACM CIKM. CIKM '02 (2002)
6. Mourão, A., Martins, F., Magalhães, J.: NovaSearch on medical ImageCLEF 2013. In: Working Notes of CLEF 2013. pp. 1–10 (2013)
7. Nguyen, D., Demeester, T., Trieschnigg, D., Hiemstra, D.: Federated search in the wild. In: Proceedings of the 21st CIKM. CIKM '12 (2012)

8. Shaw, J.A., Fox, E.A.: Combination of multiple searches. In: The Second Text REtrieval Conference (TREC-2. pp. 243–252 (1994)
9. Vogt, C., Cottrell, G.: Fusion via a linear combination of scores. Information Retrieval 1(3), 151–173 (1999)
10. Zhang, M., Song, R., Lin, C.: Expansion-based technologies in finding relevant and new information: Thu trec2002 novelty track experiments. In: TREC 2002. pp. 586–590 (2002)