Utilizing Inter-Document Similarities in Federated Search

Savva Khalaman
savvakh@tx.technion.ac.il

Oren Kurland
kurland@ie.technion.ac.il

Faculty of Industrial Engineering and Management
Technion — Israel Institute of Technology
Haifa 32000, Israel

ABSTRACT

We demonstrate the merits of using inter-document similarities for federated search. Specifically, we study a results-merging method that utilizes information induced from clusters of similar documents created *across* the lists retrieved from the collections. The method significantly outperforms state-of-the-art results merging approaches.

Categories and Subject Descriptors: H.3.3 [Information Search and Retrieval]: Retrieval models

General Terms: Algorithms, Experimentation

Keywords: inter-document similarities, federated search

1. INTRODUCTION

Federated search is the task of retrieving documents from multiple (possibly non-overlapping) collections in response to a query [1]. The task is typically composed of three steps: attaining resource (collection) description, selecting resources (collections), and merging the results retrieved from the selected collections [1]. We focus on the results-merging step; specifically, we study the merits of using information induced from inter-document similarities.

While there is much work on utilizing inter-document similarities for the single-corpus retrieval setting, there is little work along that venue for federated retrieval. For example, clustering was used to transform a single-collection retrieval setting into that of multiple collections [8]. Clusters of sampled documents were used for performing query expansion in federated search [5]; yet, inter-document similarities were not used for results merging. Furthermore, it was shown that among the clusters created across the lists retrieved from different collections there are some that contain a high percentage of relevant documents [3]; still, a results merging method exploiting these clusters was not proposed

The only work, to the best of our knowledge, that uses inter-document-similarities for (direct) results merging is based on scoring a document by its similarity with other documents in the retrieved lists [6]. We show that the method we study substantially outperforms this approach.

The method we present for merging results in federated search is adapted from recent work on fusing lists that were retrieved from the *same collection* [4]. In contrast to the non-overlapping collections setting we explore here, the retrieved lists in this work [4] were (partially) overlapping and the merging (fusion) methods used to assign initial scores to documents exploited this overlap. The adapted method that we study integrates retrieval scores assigned by a state-of-the-art results-merging approach (e.g., CORI [1] and SSL [7]) with information induced from clusters created from similar documents across the retrieved lists. Specifically, a document can provide relevance-status support to documents in the same list or in other lists that it is similar to. The resultant retrieval performance is substantially better than that of using only the initial retrieval scores assigned by the state-of-the-art merging approach.

SIGIR'12, August 12–16, 2012, Portland, Oregon, USA.
ACM 978-1-4503-1472-5/12/08.

2. RESULTS-MERGING METHOD

Suppose that some resource (collection) selection method was applied in response to query q [1]. We assume that document lists were retrieved from the selected (non-overlapping) collections and merged by *some* previously proposed results-merging algorithm [1, 7]. Let $D_{init}^{[n]}$ denote the list of the n documents most highly ranked by the merging algorithm that assigns the (initial) score $F_{init}(d;q)$ to document d.

Our goal is re-ranking $D_{init}^{[n]}$ to improve ranking effectiveness. (Documents not in $D_{init}^{[n]}$ remain at their original ranks.) To that end, we study the utilization of inter-document similarities by adapting a method proposed in work on fusing lists retrieved *from the same corpus* [4]. Let Cl be the set of document clusters created from $D_{init}^{[n]}$ using some clustering algorithm; c will denote a cluster. Then, the score assigned to document d ($\in D_{init}^{[n]}$) by the **Clust** method is:

$$F_{Clust}(d;q) \stackrel{def}{=} (1-\lambda)\frac{F_{init}(d;q)}{\sum_{d' \in D_{init}^{[n]}} F_{init}(d';q)} + \lambda \sum_{c \in Cl} \frac{F(c;q)}{\sum_{c' \in Cl} F(c';q)} \frac{\sum_{d_i \in c} Sim(d_i,d)}{\sum_{d' \in D_{init}^{[n]}} \sum_{d_i \in c} Sim(d_i,d')};$$

$F(c;q) \stackrel{def}{=} \prod_{d_i \in c} F_{init}(d_i;q)$; as all the clusters in Cl contain the same number of documents (see details below), there is no cluster-size bias incurred; $Sim(\cdot,\cdot)$ is the inter-document similarity measure used to create Cl; λ is a free parameter.

Thus, d is highly ranked if (i) its (normalized) initial score is high; and, (ii) it is similar to documents in clusters c that contain documents that were initially highly ranked. In other words, similar documents provide relevance-status support to each other via the clusters to which they belong. Note that if $\lambda = 0$, then cluster-based information is not used and $F_{Clust}(d;q)$ is d's normalized initial score.

			Queries: 51-100			Queries: 101-150		
			p@5	p@10	MAP	p@5	p@10	MAP
Uni	CORI	initial	.460	.420	.060	.396	.380	.054
		CRSC	.524	$.474^{i}$	.062	.376	.372	.052
		Clust	$\mathbf{.536}^{i}$	$\mathbf{.498}^{i}$	$\mathbf{.064}^{i}$	$\mathbf{.480}^{i}_{c}$	$\mathbf{.454}^{i}_{c}$	$\mathbf{.059}^{i}_{c}$
	SSL	initial	.432	.410	.060	.412	.396	.055
		CRSC	.492	.458	.062	.400	.346	.052
		Clust	**.504**	$\mathbf{.490}^{i}$	$\mathbf{.064}^{i}$	$\mathbf{.492}_{c}$	$\mathbf{.468}^{i}_{c}$	$\mathbf{.060}^{i}_{c}$
Rel	CORI	initial	.424	.384	.059	.432	.370	.064
		CRSC	.428	.402	.061	.456	.442	$.072^{i}$
		Clust	**.476**	$\mathbf{.442}^{i}$	$\mathbf{.063}^{i}_{c}$	**.496**	$\mathbf{.466}^{i}$	$\mathbf{.073}^{i}$
	SSL	initial	.412	.384	.095	.440	.376	.095
		CRSC	.412	.404	.096	.452	.436	$.103^{i}$
		Clust	$\mathbf{.480}^{i}_{c}$	$\mathbf{.454}^{i}_{c}$	$\mathbf{.099}^{i}_{c}$	**.492**	$\mathbf{.474}^{i}$	$\mathbf{.105}^{i}$
NRel	CORI	initial	.452	.446	.064	.416	.394	.054
		CRSC	.476	.476	.066	.472	.414	.057
		Clust	$\mathbf{.552}^{i}_{c}$	$\mathbf{.516}^{i}_{c}$	$\mathbf{.069}^{i}_{c}$	$\mathbf{.500}^{i}$	**.450**	$\mathbf{.059}^{i}_{c}$
	SSL	initial	.464	.448	.065	.396	.398	.055
		CRSC	.488	.462	.067	.456	.412	.058
		Clust	$\mathbf{.536}^{i}$	$\mathbf{.516}^{i}_{c}$	$\mathbf{.070}^{i}_{c}$	$\mathbf{.504}^{i}$	**.452**	$\mathbf{.060}^{i}$
Rep	CORI	initial	.428	.408	.060	.428	.388	.050
		CRSC	.416	.418	.060	.432	.424	.052
		Clust	$\mathbf{.492}_{c}$	$\mathbf{.474}^{i}_{c}$	$\mathbf{.064}^{i}_{c}$	**.496**	$\mathbf{.474}^{i}$	$\mathbf{.056}^{i}_{c}$
	SSL	initial	.444	.408	.061	.452	.408	.052
		CRSC	.448	.434	.060	.396	.418	.052
		Clust	**.476**	$\mathbf{.470}^{i}$	$\mathbf{.064}^{i}_{c}$	$\mathbf{.508}_{c}$	$\mathbf{.484}^{i}_{c}$	$\mathbf{.058}^{i}_{c}$

			Queries: 201-250		
			p@5	p@10	MAP
KM	CORI	initial	.468	.402	.135
		CRSC	.484	.396	.138
		Clust	**.496**	$\mathbf{.460}^{i}_{c}$	$\mathbf{.147}^{i}$
	SSL	initial	.480	.412	.152
		CRSC	.512	.412	.150
		Clust	**.532**	$\mathbf{.478}^{i}_{c}$	$\mathbf{.165}^{i}_{c}$

Table 1: Results. Boldface: the best result per testbed, evaluation measure, and initial merging method; 'i' and 'c': statistically significant differences with initial and CRSC, respectively.

3. EVALUATION

We conducted experiments with testbeds that are commonly used in work on federated search [1, 2, 5, 8, 7]: (i) **Uni**: Trec123-100col-bysource (Uniform), (ii) **KM**: Trec4-kmeans (K-means), (iii) **Rep**: Trec123-2ldb-60col (Representative), (iv) **Rel**: Trec123-AP-WSJ-60col (Relevant), and (v) **NRel**: Trec123-FR-DOE-81col (non-relevant). Titles of the TREC topics 51-150 served for queries for all testbeds except for KM where the description fields of TREC topics 201-250 were used. Tokenization, Porter stemming, and stopword removal were applied using the Lemur toolkit (www.lemurproject.org), which was used for experiments. To acquire resource (collection) description, we adopt the query-based sampling method from [2] which was also used in [1, 7, 5]. Following common practice [1, 7], the 10 highest ranked collections are selected in the resource-selection phase using CORI's resource selection method. As in previous report [7], 1000 documents are retrieved from each selected collection using the INQUERY search engine. Then, the retrieved lists are merged using either CORI's merging method [1] or the single-model SSL merging approach [7]. The **initial** score, $F_{init}(d;q)$, assigned to d by these methods is used in our Clust method.

To cluster $D^{[n]}_{init}$, we use a simple nearest-neighbors-based clustering approach [4]. For each $d \in D^{[n]}_{init}$ we create a cluster that is composed of d and the $\delta - 1$ ($\delta = 5$) documents d' in $D^{[n]}_{init}$ ($d' \neq d$) that yield the highest $Sim(d,d') \stackrel{def}{=} \exp\left(-D(p^{[0]}_{d}(\cdot)||p^{[\mu]}_{d'}(\cdot))\right)$; $p^{[\mu]}_{x}$ is the Dirichlet-smoothed unigram language model induced from x with the smoothing parameter μ (= 1000); D is the KL divergence; the term-counts statistics used for smoothing language models is based on the query-based sampling mentioned above.

The **initial** ranking induced by the CORI and SSL results-merging methods serves for a baseline. Additional reference comparison that we use is the Cross Rank Similarity Comparison (**CRSC**) approach [6] that re-ranks $D^{[n]}_{init}$ here by scoring d with $\sum_{d' (\neq d) \in D^{[n]}_{init}} \frac{Sim(d',d)}{\sum_{d'' (\neq d') \in D^{[n]}_{init}} Sim(d',d'')}$. Our Clust method incorporates two free parameters: n ($\in \{10, 30, 50, 100\}$), the number of documents in $D^{[n]}_{init}$, and λ ($\in \{0, 0.1, \ldots, 1\}$), the interpolation parameter; CRSC only depends on n. We set the free-parameter values for each method using leave-one-out cross validation performed over the queries per testbad; MAP@1000 serves as the optimization criterion in the learning phase. In addition to MAP, we also present p@5 and p@10 performance numbers. Statistically significant differences in performance are determined using the two-tailed paired t-test at a 95% confidence level.

Results and Conclusions. We see in Table 1 that Clust always outperforms the initial ranking that was induced by a state-of-the-art results-merging method; often, the improvements are statistically significantly. This finding attests to the merits of integrating the initial results-merging score with information induced from clusters of similar documents. Further exploration reveals that $\lambda \notin \{0, 1\}$ often yields optimal performance. This shows that the integration just mentioned yields performance that is better than that of using each of the two integrated components alone. We also see that Clust consistently outperforms CRSC, which does not utilize the initial results-merging scores, nor uses a cluster-based approach.

Acknowledgments We thank the reviewers for their comments. This paper is based upon work supported in part by the Israel Science Foundation under grant no. 557/09. Any opinions, findings and conclusions or recommendations expressed here are the authors' and do not necessarily reflect those of the sponsors.

4. REFERENCES

[1] J. Callan. Distributed information retrieval. In W. Croft, editor, *Advances in information retrieval*, chapter 5, pages 127–150. Kluwer Academic Publishers, 2000.

[2] J. Callan and M. Connell. Query-based sampling of text databases. *ACM Transactions on Information Systems*, 19(2):97–130, 2001.

[3] F. Crestani and S. Wu. Testing the cluster hypothesis in distributed information retrieval. *Information Processing and Management*, 42(5):1137–1150, 2006.

[4] A. Khudyak Kozorovitsky and O. Kurland. Cluster-based fusion of retrieved lists. In *Proceedings of SIGIR*, pages 893–902, 2011.

[5] M. Shokouhi, L. Azzopardi, and P. Thomas. Effective query expansion for federated search. In *Proceedings of SIGIR*, pages 427–434, 2009.

[6] X. M. Shou and M. Sanderson. Experiments on data fusion using headline information. In *Proceedings of SIGIR*, pages 413–414, 2002.

[7] L. Si and J. Callan. A semisupervised learning method to merge search engine results. *ACM Transactions on Information Systems*, 21(4):457–491, October 2003.

[8] J. Xu and W. B. Croft. Cluster-based language models for distributed retrieval. In *Proceedings of SIGIR*, 1999.