RESEARCH PAPER | Computer Science | Volume : 5 | Issue : 4 | April 2015 | ISSN - 2249-555X

On Score Normalization in Distributed Information Retrieval

KEYWORDS: Score distribution ; Result merging ; Normalization ; Distributed information retrieval; Data Fusion ; Information Retrieval

Benjamin Ghansah
School of Computer Science and Communication Engineering, Jiangsu University, 301 Xuefu Road 212013 Zhenjiang, Jiangsu, China

Benuwa Ben-Bright
School Of Computer Science, Data Link Institute P. O Box 2481 Tema Ghana, West Africa.

ABSTRACT *This paper presents a novel result merging technique that uses the idea of score normalization for curve fitting. First the algorithm computes the cumulative score distribution from an ideal distribution: we used the central sample database for this purpose. Second we used the exponential and Gaussian distributions to build an optimal score distribution (OSD). Third we normalize the scores returned by individual information source by mapping each score to the OSD. Lastly the normalized values obtained are used as global scores for result merging. We compare our results with SAFE merging algorithm, which showed superior performance on two testbeds; TREC123, and TREC4kmeans.*

1 INTRODUCTION

Prior results merging algorithms such as CORI [9] and SSL[27] make use of document relevance scores returned from the information sources in order to work effectively. These scores serves as a measure of the degree of relevance to an input request and are subsequently used in ranking retrieved items. However, in most contemporary information retrieval settings, information sources return only ranked lists of documents without scores. This is largely because direct interpretation of the scores returned with documents is practically vague to an average user. For example, a difference of 0.1 in ranking (i.e. 0.6 and 0.5) may be due to a difference in score,but this difference (i.e. 0.1) does not suggests that the document with score 0.6 is 0.1 more relevant than the document with score 0.5 beneath it.[4]however suggested that algorithms that rely on relevance scores in order to function could use pseudoscores (assigning artificial relevance scores) to the returned documents in a heuristic manner. It should be noted however that, in most practical situations—where document scores are not available, assigning pseudoscores to documents produce poor results, hence algorithms that do not rely on returned documents scores, especially in non-cooperative environments is highly recommended than the former.

A major problem with both rank-based and score-based aggregation algorithms is to calculate a global score for each document,that is comparable to the scores of documents returned by other information sources. In order to combine the scores produced by different sources, the scores should be first made comparable across input systems[11]:this is usually accomplished by a normalization phase[21].One possible reason for having disparities in sources' scores may be the use of differing retrieval strategies (e.g., probability of relevance ,vector space or p-norm Boolean), or information sources having different lexicon statistics. For example,[23] observed that scores' range and distribution varies wildly across different models making them incomparable across different information sources, thus the document scores or ranks returned by multiple information sources are not directly comparable, making them incongruous for merging.In ad hoc retrieval systems, the diversity of score types is not an issue; scores do not have to be comparable across different systems; they are only used to rank documents per request per information source. It could be argued that, ad hoc retrieval systems in contemporary times also merges other forms (images,videos,maps etc.) of information alongside the conventional text output. However that is not the focus of this paper.

In advanced applications, such as distributed retrieval, data fusion or meta search,some form of score normalization is inevitable. In these applications, several rankings have to be merged or fused to a single ranked list to be presented to the user. According to [7], score normalization is an important step in achieving an effective rank list from varied source ranks. However, [12] observed that performance of rank-based or score-based aggregation may be affected by artificial, usually redundant nonconformities consistently occurring in the input score distributions, which does not affect the performance of each ranking technique separately, but distort the collective result when the individual biases differ from each other, and therefore it should be possible to improve the results by preventing or mitigating these deviations.It is important to note that, traditional approaches such as range normalization based on minimum and maximum scores are rather naive, considering the enormous disparity of score outputs across information sources: they do not take into account the shape of score distributions. Although these methods have worked reasonably well for merging or fusing results,e.g. [19], advanced approaches have tried to improve normalization by investigating each candidate information source scores [2], and has proved significantly better in terms of performance than as proved latter.

There are two environments in a Distributed Information Retrieval (DIR): Cooperative and non-cooperative environment. In a cooperative environment,collections return their complete lexicon statistics such as; term frequency, document frequency, term weight, and document weight of each returned answer to the broker, based on an agreed protocol existing between collections commonly known as STARTS [15]. Due to the cooperation that exist among collections and the availability of attributes of each collection to the broker, documents are merged according to their calculated similarities based on the received statistics by

Volume : 5 | Issue : 4 | April 2015 | ISSN - 2249-555X

the broker. Though returned documents from the individual collections are not comparable, enough information is available to attain a realistic normalization and subsequent merger.

In an uncooperative environment, information sources provide brokers not more than a search interface, and are also assumed to return only a list of documents, without similarity scores or other such information. A technique commonly used is to estimate candidate sources' statistics,by creating a resource description or collection summaries of the collections [7]. Collection summaries can be provided by the query-based sampling technique. In this technique an initial query is selected from a list of common frequent terms (e.g. from a reference dictionary) and is submitted to the collection. A few of the documents returned for the initial query are downloaded. The next query is selected from the text of the downloaded documents, and the process repeats. The sampling stops once a sufficient number of documents have been downloaded from each collection. The downloaded documents are used to form a central sample database(CSD).

According to [26], the documents downloaded by query-based sampling may not be a good random sample of the available information sources, creating mismatches in the distribution of sampled documents. [33] attributed this to the varied biases that exist in candidate information sources' retrieval models, or the query sets used.[25] however argued that though the assumption of randomness is questionable, the accuracy of estimated scores is rather acceptable. Based on this, we assume that documents sampled from the individual sources in order to create the CSD are uniformly distributed in the total ranking from the originated information sources. The aim of this study is to utilize and implement score normalization techniques in the result merging stage of DIR. We focus on the exponential and Gaussian distributions model, by investigating the theoretical as well as the empirical evidence supporting its use. We are studying the result merging phase it is very important stage in the overall performance of the DIR system. For example, in situations where the most suitable information sources have been chosen in the preceding stages, if the merging is not effective, the general quality of the retrieval process will be suboptimal. This importance is amplified particularly in the web environment where users rarely look past the top 20 results and most often do not browser after the top 5 results [17].The rest of the paper is organized as follows: We will discuss the related work in Section 2 and present our proposed framework for the result merging problem in DIR in Section 3. Sections 4 describe the experimental setup. Section 5 presents the experimental results. Finally, Section 6 presents our concluding remarks and future directions.

2 RELATED WORK

Considerable research have been conducted for all three areas of DIR as resource representation, result selection and result merging [7, 24]. This section provides a discussion of prior research on result merging, as well as a brief review on resource representation and resource selection. We also discuss some popular Score Normalization methods used in ad hoc search.

Federated search, or distributed information retrieval[7, 24], is the problem of automatically searching across multiple distributed collections or resources. Distributed Information Retrieval research is divided into three separate, but interrelated, tasks. *Resource representation;*the task of soliciting lexicon information about the contents of each candidate information sources. *Resource selection;* the task of deciding which information source to search for a given query. *Results merging*; the task of merging results from different information sources—those selected—into a single document ranking. It is often not feasible and usually computationally expensive to issue the query to every available information source. Therefore, the goal of resource selection is to determine which subset of the available resources are most likely to have relevant content. Resource representation happens off-line, while resource selection and results merging happens every time a query is issued to the system.

As mentioned in the previous section, there are two environments in a DIR ; *cooperative and uncooperative* environment. In a cooperative environment, resources provide the system with all the information needed to perform an effective and efficient DIR. A cooperative environment may occur, for example, when the system and its target resources are operated by the same search company. In an uncooperative environment, resources provide the system no more than the functionality they provide their human users: a search interface. An uncooperative environment may occur, for example, when the system searches across external digital libraries that are located in the *deepweb* (part of the web that cannot be crawled or harvested).In a cooperative environment, DIR can be achieved with an agreed protocol such as STARTS [15]. The STARTS protocol standardizes how resources should i) publish their content descriptions, ii) define a unified query language to retrieve documents from resources, and iii) specifies result set statistics to be provided alongside search results to facilitate results merging.

Resource selection is the next step in DIR after the *broker* has obtained basic lexicon statistics (descriptions or summaries) of available information sources. Some methods rank sources by comparing the text in the query with the text in the *entire* information source, using metrics adapted from the prevailing document retrieval system. These techniques model the sources as a single unit of retrieval and assume there are no peculiar difference(in terms of size, term frequencies etc.) between documents in the information source. For this reason, they are occasionally referred to as "large document models". CORI, proposed by [9] falls under this category. The technique implements the INQUERY inference network approach to resource selection. [28] proposed a new source selection method-ReDDE, which estimates the number of relevant documents in information sources according to their sampled documents. Information sources are ranked according to the number of their sampled documents that are ranked highly by a central model. Their method significantly outperformed CORI in most settings. [22] introduced a decision theoretic framework (DTF) for resource selection. It tries to minimize the overall costs of DIR including money, time, and retrieval quality. However, the effectiveness of DTF, in particular for short queries, has been found to be inferior to that of CORI [22]. [30] proposed a unified utility maximization framework (UUM) for resource selection. UUM runs queries on an index of all sampled documents. It uses training queries to learn the probabilities of relevance for the sampled documents according to their central scores. Using these probabilities, UUM selects collections that are likely to maximize either the final precision or final recall. Hawking and Thomas [2005] suggested a hybrid approach that combines federated search with centralized techniques. In their method, the link anchor text available in a

Volume : 5 | Issue : 4 | April 2015 | ISSN - 2249-555X

set of crawled documents is used to provide a description of collections that are not crawled. Information sources are ranked according to the similarities of crawled pages referring to them. They showed that their technique can outperform ReDDE and CORI for some tasks.

In recent times, learning based models have also been proposed for resource selection. They treat resource selection in DIR[16] as a classification problem. In particular, given a set of training queries and some relevance judgment, a classification model can learn to predict the relevance of an information source. In some settings, the classification methods have been shown to provide more precise resource selection results than prior methods without the training process.

Ghansah and Benuwa [14] proposed DelCosim, a collection selection approach which addresses the issue of duplicate collections in DIR, they used a local fingerprint method to identify and remove a duplicate pair with a minimum size. Their technique achieved a more diversified results output than prior methods.

Results merging is the task of combining results from different resources—those selected from the resource selection—into a single unified ranked list. Even when resources adapt a similar retrieval algorithm, they often use different representations (e.g., stemming) and have different corpus statistics (e.g., *tf* values). For these reasons, documents scores (for the same query) from different sources may not be directly comparable across resources. To addressthese problems, the goal of results merging is to perform score normalization. That is,to transform each retrieved document's score into a general score which is comparable across resources. Results from different resources can then be ranked based on their normalized scores (derived general score). Prior result merging methods can be categorized into two groups. The first group assumes some level of collaboration of distributed information sources and use some statistics provided by those sources for merging [7, 18]. In environments where information sources are uncooperative, a semi-supervised learning (SSL) [29] method is a more practical approach. In this technique, after the query-based sampling stage [7], each information source is represented by a set of sample documents. A collection of all the sampled documents is referred to as the centralized sample database. SSL uses the overlapping documents in both individual ranked lists and centralized sample database to construct a regression model. Once regression is done, SSL can convert the rank of any document returned from an individual source to that document's centralized score. These centralized scores are used as global scores to merge all other documents. [25] proposed a novel result merging method known as the Sample-Agglomerate Fitting Estimate (SAFE) which do not rely on overlap documents between individual sources and the CSD. Instead, the technique estimates document ranks based on the uniform sampling assumption, and uses those estimated ranks for regression. SAFE method however, does not distinguish the contribution of overlapping documents with accurate ranks (i.e., existing in the source's returned list) and sample documents with estimated ranks for regression. Evaluation of the performance of SAFE against state-of-the-art approaches showed superior results.

Based on the intuition of binary relevance (relevance or non-relevance) in relation to query-document study, standard attempt to model score normalizations on per-request basis, as a combination of two distributions: one for relevant and the other for non-relevant documents [5, 6, 20, 31, 32]. Given the two constituent distributions and their amalgam weight, the probability of relevance of a document given its score can be computed directly, primarily ensuring the normalization/ standardization of scores into probabilities of relevance [3, 20]. Additionally, the anticipated numbers of relevant and non-relevant documents, score or diversity measure can be conveniently estimated, facilitating the computation of precision, recall, or any other standard measure at any given threshold enabling its optimization [3].Theoretically, the right combination of components in such methods yields a somewhat *clean* and non-parametric output.

There has been numerous groupings of distributions proposed since the inception of IR— two normal of equal variance [31], two exponential [32], two Poisson [6], two gamma [5], normal for relevant and an exponential for non-relevant[1, 3, 10, 20, 34]. A recent attempt by [12, 13] to model score normalizations without reference to relevance seems to overcome some of the practical issues of mixture models. Their model aggregate score normalizations of many requests, on per-engine basis,with single distributions; this enables normalization of scores to probabilities—though not of relevance—comparable across different engines. The approach was found to perform better than the simple approaches in data fusion environments. Note that one possible approach to modeling score normalizations is to first convert the scores into some form which exhibits better distributional attributes. In principle, any monotonic transformation of the scores produced by a reliable system would suffice for this method. Thus one might for example transform a score which appeared to give a lognormal distribution for some relevance collection, into one which gave a probability of relevance, by taking the log of the score.

3 PROPOSED ALGORITHM

The algorithm presented, although influenced by the work by [12], differs from their approach considerably in that, it does not rely on the Central Sample Database created from candidate collections. Their algorithm assumes the availability of an optimal score distribution (OSD) defined as the score distribution of an ideal scoring function that matches the ranking by actual relevance. Again their work was conducted in an ad hoc search environment.

Let U be the universe of information objects to be ranked, and L the set of rank lists to be combined. Each rank source $\lambda \in L$ can be represented as a one-to-one correspondence function $\lambda : U_\lambda \rightarrow$ for some $U_\lambda \subset U$, where for each $x \in U_\lambda$, $\lambda(x)$ is the position of x in the ranking returned by λ. For each $\lambda \in L$, we shall denote by $S\lambda$: $U \rightarrow L$ the scoring function associated to λ, where we take $S_\lambda(x) = 0$ if $x \notin U\lambda$.

The approach consists of two phases. The first phase is performed offline, as follows:

1. For each ranked list $\lambda \in L$, compute the cumulative score distribution G_λ of the values S_λ returned by the information source that outputs λ. This can be approximated by the sampled documents in the CSD emanating from the specific information source.

2. The CSD is used to build a so called optimal score distribution by using the exponential and Gaussian distributions, proposed by Manmatha et al [4] i.e. : $[0,1] \rightarrow [0,1]$.

Volume : 5 | Issue : 4 | April 2015 | ISSN - 2249-555X

3. Normalization: For each $x \in U$ and $\lambda \in L$, map the score of each rank source to the OSD:

In this technique, the normalization stage preserves the order of each source rank list, with the exception of situations where the values remains the same (i.e. score value unlikely to fall), since preserves the given rank order. The resulting scores range in [0,1], and their distribution is for all $\lambda \in L$, thus eradicating any potential biases or noise.

4. Combination: merge the normalized scores, e.g. by a linear combination or some other score-based technique.

4 EXPERIMENTAL SETUP

Datasets

Our experiments was conducted with two TREC datasets: trec123 - which contains 100 collections of TREC CDs 1, 2 and 3 organized by publication sources; and trec4-kmeans, which contains 100 collections created from TREC4 by k-means clustering[7]. We assign different retrieval models to each information source in a round robin fashion. The retrieval models used includes language modeling, vector space tf-idf and INQUERY[8]. 200 documents were sampled from each information source and we used ReDDE to select the top 5 sources for each query and INQUERY for querying the centralized sample database.

Baseline methods

We compared our result with SAFE, as it is believed to be one of the best result merging techniques. We used the hybrid function *f*, which was shown by [25] to produce the best results.

5 EXPERIMENTAL RESULTS

Table 1: Document Precision on TREC123, TREC4kmeanswith top 30 Documents of each Source

	TREC123					TREC4kmeans				
P@n	@5	@10	@15	@20	@30	@5	@10	@15	@20	@30
SAFE	0.33	0.28	0.26	0.23	0.21	0.27	0.24	0.21	0.19	0.14
SD	0.36	0.34	0.30	0.28	0.26	0.31	0.28	0.26	0.23	0.20

Table 1 shows the precision results when top 30 documents returned from each source are merged. In general, our proposed method (SD) constantly outperforms SAFE. For trec4- kmeans dataset, the precision increases when more documents from each source are combined. However, that was different with the trec123 dataset. It may be contended that, the algorithm is susceptible to noisy data when merging more documents.One possible solution is to assign even smaller weight to documents at the bottom of the returned list.

6 CONCLUSION AND FUTURE DIRECTION

This paper proposes a result merging algorithm for Distributed Information Retrieval. We have demonstrated how to model the score distributions of a number of text search engines in a DIR environment. Specifically, it was shown empirically that the score distributions on a per query basis may be fitted using an exponential distribution for the set of non-relevant documents and a Gaussian distributions for the set of relevant documents. Empirical results on two datasets have shown the effectiveness of the proposed results merging algorithm. Future work will attempt to improve the modeling for better performance and use other combination approaches for relevance and non-relevance documents.

REFERENCE

1. Arampatzis, A. Unbiased sd threshold optimization, initial query degradation, decay, and incrementality, for adaptive document filtering. in TREC. 2001. | 2. Arampatzis, A. and J. Kamps. A signal-to-noise approach to score normalization. in Proceedings of the 18th ACM conference on Information and knowledge management. 2009. ACM. | 3. Arampatzis, A. and A. van Hameran. The score-distributional threshold optimization for adaptive binary classification tasks. in Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval. 2001. ACM. | 4. Avrahami, T.T., L. Yau, L. Si, and J. Callan, The FedLemur project: Federated search in the real world. Journal of the American Society for Information Science and Technology, 2006. 57(3): p. 347-358. | 5. Baumgarten, C. A probabilistic solution to the selection and fusion problem in distributed information retrieval. in Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. 1999. ACM. | 6. Bookstein, A., When the most "pertinent" document should not be retrieved—an analysis of the swets model. Information Processing & Management, 1977. 13(6): p. 377-383. | 7. Callan, J., Distributed information retrieval, in Advances in information retrieval. 2000, Springer. p. 127-150. | 8. Callan, J.P., W.B. Croft, and S.M. Harding. The INQUERY retrieval system. in Database and expert systems applications. 1992. Springer. | 9. Callan, J.P., Z. Lu, and W.B. Croft. Searching distributed collections with inference networks. in Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval. 1995. ACM. | 10. Collins-Thompson, K., P. Ogilvie, Y. Zhang, and J. Callan. Information Filtering, Novelty Detection, and Named-Page Finding. in TREC. 2002. | 11. Croft, W.B., Combining approaches to information retrieval, in Advances in information retrieval. 2000, Springer. p. 1-36. | 12. Fernández, M., D. Vallet, and P. Castells, Probabilistic score normalization for rank aggregation, in Advances in Information Retrieval. 2006, Springer. p. 553-556. | 13. Fernández, M., D. Vallet, and P. Castells. Using historical data to enhance rank aggregation. in Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval. 2006. ACM. | 14. Ghansah, B. and B.-B. Benuwa, Fingerprint Based Approach for Resource Selection in Federated Research International Journal of Advanced Research in Computer Science & Technology (IJARCST) 2014. 2(3): p. 329-333. | 15. Gravano, L., C.-C.K. Chang, H. García-Molina, and A. Paepcke, STARTS: Stanford proposal for Internet meta-searching. Vol. 26. 1997: ACM. | 16. Hong, D., L. Si, P. Bracke, M. Witt, and T. Juchcinski. A joint probabilistic classification model for resource selection. in Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval. 2010. ACM. | 17. Jansen, B.J., A. Spink, and T. Saracevic, Real life, real users, and real needs: a study and analysis of user queries on the web. Information processing & management, 2000. 36(2): p. 207-227. | 18. Kirsch, S.T., Document retrieval over networks wherein ranking and relevance scores are computed at the client for multiple database documents. 1997, Google Patents. | 19. Lee, J.H. Analyses of multiple evidence combination. in ACM SIGIR Forum. 1997. ACM. | 20. Manmatha, R., T. Rath, and F. Feng. Modeling score distributions for combining the outputs of search engines. in Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval. 2001. ACM. | 21. Montague, M. and J.A. Aslam. Relevance score normalization for metasearch. in Proceedings of the tenth international conference on Information and knowledge management. 2001. ACM. | 22. Nottelmann, H. and N. Fuhr, Decision-theoretic resource selection for different data types in MIND, in Distributed Multimedia Information Retrieval. 2004, Springer. p. 43-57. | 23. Robertson, S., On score distributions and relevance. 2007: Springer. | 24. Shokouhi, M. and L. Si, Federated search. Foundations and Trends in Information Retrieval, 2011. 5(1): p. 1-102. | 25. Shokouhi, M. and J. Zobel, Robust result merging using sample-based score estimates. ACM Transactions on Information Systems (TOIS), 2009. 27(3): p. 14. | 26. Shokouhi, M., J. Zobel, F. Scholer, and S.M. Tahaghoghi. Capturing collection size for distributed non-cooperative retrieval. in Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval. 2006. ACM. | 27. Si, L. and J. Callan. Using sampled data and regression to merge search engine results. in Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval. 2002. ACM. | 28. Si, L. and J. Callan. Relevant document distribution estimation method for resource selection. in Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval. 2003. ACM. | 29. Si, L. and J. Callan, A semisupervised learning method to merge search engine results. ACM Transactions on Information Systems (TOIS), 2003. 21(4): p. 457-491. | 30. Si, L. and J. Callan. Unified utility maximization framework for resource selection. in Proceedings of the thirteenth ACM international conference on Information and knowledge management. 2004. ACM. | 31. Swets, J.A., Information retrieval systems. Science, 1963. 141(3577): p. 245-250. | 32. Swets, J.A., Effectiveness of information retrieval methods. American Documentation, 1969. 20(1): p. 72-89. | 33. Thomas, P. and D. Hawking. Evaluating sampling methods for uncooperative collections. in Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval. 2007. ACM. | 34. Zhang, Y. and J. Callan. Maximum likelihood estimation for filtering thresholds. in Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval. 2001. ACM.