SURVEY ON SCORE NORMALIZATION : A CASE OF RESULT MERGING IN DISTRIBUTED INFORMATION RETRIEVAL

BENJAMIN GHANSAH , SHENGLI WU
School of Computer Science and Communication Engineering
Jiangsu University
301 Xuefu Road 212013 Zhenjiang, Jiangsu
China
ben@datalink.edu.gh, swu@ujs.edu.cn

Abstract: - Merging the outputs of different search engines or information sources in response to a query has been shown to improve performance. In most cases, scores produced by different information sources are not comparable: merging techniques are often segregated into a score normalization step followed by a combination step. The Combination step is usually straight forward and has been an area of active research. However, the normalization step has received less attention; in particular a peculiar attribute such as *diversification* is largely missing in most Result Merging studies. This survey seeks to explore the various domains of score normalization, especially the results merging phase of a Distributed Information Retrieval environment, and propose a general framework to diversify score normalization through the use of the covariance principle.

Key-Words: - Machine learning , Classification approach , Distributed Information retrieval , Result merging , Information Retrieval

1 Introduction

Studies has showed that merging outputs of various search engines or information sources to produce a unified (combined) ranked list in response to a user query, can significantly improve search performance[[23],[5],[22],[19]].Generally speaking, due to the difference in algorithms and lexicon statistics, relevance scores produced by information sources are not comparable. *Score normalization* is concerned with making document relevance scores, produced by multiple information sources comparable. There are three concepts in Information Retrieval that usually use score normalization in order to fuse documents from different retrieval models. They include Result merging, Metasearch and Data fusion. These concepts are similar in the sense that, they all combine documents from multiple retrieval systems and aggregate them in to a unified final list. However, they differ in the techniques in implementing the combination .

Result merging in Distributed Information Retrieval(DIR) aims at combining top-ranked results returned for a query by different information sources into a single list. Most DIR merging techniques assume that the rate of overlap among information sources is either none or insignificant. Note that Result merging is preceded by two important phases i) Source Representation - where lexicon statistics also known as summaries of the available information sources are derived and ii) resource Selection–given a user query and summaries of the available information sources, suitable subset of candidate sources would be selected to answer a user query. This is because it is usually not feasible to have an exhaustive resource selection due to cost, in terms of bandwidth and latency.

In data fusion methods, documents in a single information source are ranked with different retrieval models. The main aim of data fusion is to produce a single accurate ranking list from the ranking lists of different retrieval models. There are no resource representation sets and no resource selection.[[38] , [23] ,[30] , [6], [37]] as in the case of result merging.

Metasearch merging technique receives results returned by multiple search engines with overlapping indexes and combined them into a single ranked list. In Metasearch merging, voting plays a central role for computing the final rank of a document. For example, documents returned by many search engines are expected to be highly ranked in the final combined list. In the situation where there is no overlap between the results, majority of Metasearch merging techniques cease to function. A typical example is SavvySearch [15] . The discussions so far shows that, each merging technique relies on score normalization(or standardization of scores) in order to have a comparable score for easy merger. The contribution of this article can be summarized in some main points:

1. Review of Result merging, data fusion and Metasearch score normalization techniques.
2. We propose a general framework to improve Score normalization
3. We list different challenging and promising research directions.

Note, *information sources* and *sources* are used to refer to search engines and databases(repositories or collections) respectively in this paper.

The rest of the article is structured as follows. In the next section, section 2, we review result merging , data fusion and Metasearch . Then, in Section 3, we present an overview of existing approaches in score normalization and provide a general framework to improve score normalization . In the last section-section 4, we provide conclusions and future research directions.

2 Related work

2.1 Result Merging

Si and Callan [32] defined result merging as combining multiple result lists into a single unified ranked list. Individual result's-list are obtained by sending the same user query to an N number of different information sources. In a Distributed Information Retrieval environment, sources may use different retrieval algorithms and varied lexicon statistics. Thus, the document scores or ranks returned by multiple collections are not directly comparable and are not consistent for merging. Result merging algorithms aims at calculating a universal score for each document that is comparable to the scores of documents returned by other information sources. Note in a DIR environment, there are no restrictions on the overlap-rates between the document collections of the different information sources, nor are there any restrictions on what ranking functions should be used. The results merging phase is the last step of the distributed information retrieval process, where the individual result lists from the remote sources are merged into a single unified list, which is returned by the DIR system to the user. Studies by Callan [1] and Craswell, Hawking [4] reported that results merging phase is critical to the overall performance of the retrieval process, typically in precision oriented environments where users expect a significant number of relevant documents in the top ranks of the returned document list. Studies have shown that retrieval processes will be suboptimal or ineffective when the best relevant information sources are selected to answer a query and a less effective merging technique is applied to get a unified merged list. Jansen, Spink [16] observed that, putting more effort in ensuring good results output with high precision is essential in satisfying a user's search experience.

The main feature of result merging is that, it uses resource descriptions to perform score normalization and, thus, it can be treated as a semi supervised normalization approach. There are three main results merging methods developed in literature : CORI [[1],[2]], SSL [32] and SAFE [31] with its WCF modification [12] .

CORI result merging

The CORI result merging technique [[1];[2]] , relies on the aggregation of two linear components for score normalization: i. the score of the information source during source selection(specifically CORI collection selection algorithm) and ii. score of the document in a specific information source. Based on these parameters, the source scores are normalized as:

$$\acute{S} = \frac{S - S_{min}}{S_{max} - S_{min}} \qquad (1)$$

where, S is the collection selection score computed by the CORI collection selection algorithm [[1];[2]] , $\acute{S}$ denotes the normalized score ranging between [0; 1]. *Smin* and *Smax* are minimum and maximum source's score. The document scores are normalized in a similar fashion as:

$$\acute{D} = \frac{D - D_{min}}{D_{max} - D_{min}} \qquad (2)$$

D_{max} and D_{min} are the maximum and minimum document scores assigned by the information sources and $\acute{D}$ is the document normalized score. Note that D_{max} and D_{min} require cooperation from the remote collection to be set. In situations where this cooperation is not available, they are set to the relevance score achieved by the highest and the least relevant documents respectively. For a document returned with score D from a source with normalized collection selection score of S, CORI computes normalized document score as:

$$d = \frac{\acute{D}+0.4\times\acute{D}\times\acute{S}}{1.4} \qquad (3)$$

Finally, d is the final document score, which again is normalized between 0 and 1. CORI merging formula uses heuristic weighting schemes such as weight 1 for normalized document score and weight 0.4 for normalized collection selection score in Equation 3. The heuristic weighting scheme significantly limits the effectiveness of CORI merging as it may not adapt to different types of queries and information sources.

Semi-Supervised Learning (SSL)

Si and Callan [32] trained a regression model for each collection that maps document scores into their global (merging) scores. SSL assumes that the optimal merging of documents that are returned from remote collections and consequently the optimal final result list is the one that approximates as closely as possible the list that would be returned, if all the documents were available for indexing in a single centralized index. For this purpose, SSL creates a central index (CSI) of all sampled documents retrieved from available sources. For a given query, some of the documents that are returned from the sources may already be available in the central sample index. SSL runs the query against the CSI and compares the centralized scores of such overlap sampled documents. Note that the overlapping documents have both the source-specific score provided by their corresponding originating source and a normalized score, which is calculated based on position in the CSI. Based on this information, a linear regression is trained in the following form:

$$S_n(d|q) = a.s(d|q) + b \qquad (4)$$

where $s(d|q)$ and $S_n(d|q)$ are the source-specific and the normalized document scores respectively. This regression method is then used to normalize all scores from the given ranked list. The main drawback of SSL is that it requires an overlap between source-specific ranked lists and sampled documents in CSI. An alternative technique by [24] reported that some documents from a ranked list can be downloaded, indexed and ranked locally. This approach however is costly and may increase latency.

In summary, the SSL algorithm functions as follows: given an information need (query) and a number of information sources, the query is routed to the appropriate information sources and is also submitted to the CSI. The algorithm subsequently receives as input a list of documents with their corresponding source-independent scores from the CSI and a set of lists of documents with their respective source-dependent scores from the selected source. The technique assumes that some documents in the CSI *overlap* with documents in the result list of each individual source.

This algorithm utilize the advantage of the existing overlap documents between the result lists of the individual sources and the CSI and uses their corresponding relevance scores to estimate a linear model, that maps the result lists of the individual sources to the CSI. The primary aim of the algorithm is to use this estimated model, which is usually different for each information source, in order to assign source-independent scores to the non-overlap documents, thus estimating a score for every returned document.

Sample-Agglomerate Fitting Estimate (Safe)

Shokouhi and Zobel [31] proposed an algorithm that work with minimum cooperation between the broker and information sources. SAFE uses the scores of all documents in cluster of all the collection samples, and generates a statistical fit to estimate scores. In this method, the query is run on the CSI as well as on the original collections. The scores obtained in response to the query on the CSI, which are based on incomplete global statistics, are used to interpolate scores for the documents obtained in response to the query from each original collection. The technique produces an optimal results due to the fact that the scores for the CSI is able to provide fairly tight bounds and precise estimates for the scores of the returned documents, even if there is no overlap between the CSI and original documents. The great advantage of SAFE is that, it does not require document relevance scores to be provided by sources, however , He, Hong [12] observed that, SAFE 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. Further to that, top ranked documents (in source-specific list) are probably more important for curve fitting because of the goal of high-precision, which is not considered in SAFE. Based on the observation, they propose a novel result merging method called Weighted Curve Fitting (WCF) that combines the features of SSL and SAFE for result merging.The new method accurately distinguishes estimated rank information. It also considers the importance of documents in different positions for regression.

2.2 Data Fusion

Data Fusion techniques involves the merging of search results of multiple retrieval systems from a single information source. This fusion algorithm accepts two or more ranked lists and merges these lists into a single ranked list with the aim of creating a single precise ranking list from the ranking lists of different retrieval systems. Unlike the Results Merging stage in DIR, there is no resource description and no collection selection prior to the merging of ranked list. The principle of combining multiple retrieval algorithm in responds to a user query to improve results effectiveness has been extensively studied in prior research [[8]; [17] ; [13] ; [36]]. Data Fusion techniques can be broadly divided into two approaches: Supervised and Unsupervised.

Supervised data fusion techniques usually extract metadata information or lexicon statistics from candidates documents or submitted ranked list, and then employ a machine learning algorithm to train the fusion model. One possible way of employing supervised data fusion is when there is a way to control the use of information existing in categorized training data. For example Liu, Meng [21] developed a general framework for conducting supervised data fusion, in which training is ordered as an optimization problem in which one reduces inconsistencies between ranking results and the categorized data. Sheldon, Shokouhi [29] proposed k-Merge, a technique that first extract features from both the lists and the documents appearing in any of the lists, and then uses a learning rank method to optimize a given metric, like MRR or MAP, to combine the lists into a final merging list in response to a user query. Qin, Geng [25] proposed a supervised probabilistic data fusion technique, which is based on coset-permutation distance and defined in a stage-wise manner. Note that when training data is available, the effectiveness of data fusion methods using only ranks can be comparable to those that use document scores reported by the individual systems. In summary, supervised setting makes sufficient modeling assumptions and data fusion is in principle straightforward. For instance in classification, the task is to combine data sources such that the classification accuracy is improved. However ,though the task is rather well defined in supervised settings, there are still practical challenges in its implementation.

In an unsupervised data fusion environment however, it is not straightforward to define or search for agreement between documents or lists due to the absence of a clear premise. A typical technique however is to combine data sources by maximizing mutual dependencies between them such that the shared aspects between them is preserved. This kind of approach is useful in cases where the information shared by data sources is more interesting than the information specific to the data sources [34].

Fox and Shaw [10] proposed unsupervised data fusion methods including operators like the MINMAX,CombSUM and CombMNZ. CombSUM has been studied extensively in information retrieval research [[5] ; [17]; [28, 35]]. CombSUM involves setting the score of each document in the combination to the sum of the scores obtained by the individual information sources. In CombMNZ the score of each document is obtained by multiplying the sum obtained in CombSUM by the number of information sources which had non-zero score. Note that summing (CombSum) is equivalent to averaging, while CombMNZ is equivalent to weighted averaging. Lee [19] did a further study with six different information sources. They normalized each information source on a per query basis improving results substantially and showed that CombMNZ worked best, followed by CombSum while operators like MIN and MAX were the worst. Other unsupervised data fusion approaches include, Burst-aware data fusion for microblog search [20], data fusion in clustering microarray data [18], data fusion for the management of multimedia documents [7] and the outranking model for fusion, [9] ,among others.

2.3 Metasearch Merging

Metasearch algorithms are often used interchangeable with results merging task of DIR. But the meaning of metasearch techniques depend on the fact that there are multiple scores for a single document (ie. there is overlap between information sources). In contrast, in the results merging task of DIR, the contents of the information sources are usually independent and we cannot expect many documents to appear in two or more ranked lists. In metasearch merging, the results returned by multiple information sources with overlapping indexes are combined in a single ranked list. Note that many information sources produce scores as a measure of the relevance of the document to a particular user query. These scores are then used to generate document rankings. The scores produced by different search engines are usually not comparable. Montague and Aslam [23] proposed three normalization techniques for meta-search. The methods involved linearly shifting and scaling scores so that the following mappings were achieved: They tested these with some well-known combination techniques CombSum and CombMNZ [10] described in the previous section above.

A recent work by Manmatha, Rath [22] showed that the scores of non-relevant documents may be approximated by an exponential distribution and the scores of relevant documents by a Gaussian distribution, they also showed that the relevant and non-relevant distributions could be improved by solving a mixture model consisting of an exponential and a Gaussian using Expectation-Maximization (EM). They used mixture model to map scores to probabilities for each engine. The probabilities were averaged for meta-search. The results showed similar performance as the CombMNZ technique with the Standard normalization.

3 SCORE NORMALIZATION

In a Distributed Information Retrieval (DIR) system, when the broker receives a user's information need or query, it routes the query to a number of selected relevant sources. The selected sources use internal algorithms to rank their documents for the given query and return lists of ranked documents to the broker. The broker then normalizes the scores of the returned ranked lists from the individual sources in order to make them comparable and subsequently merge them into a unified list.

Existing techniques to score normalization can be classified into linear, non-linear, results merging and rank-based categories. The first three categories require document relevance scores and hence normalization is needed to make scores comparable for easy merging. The rank-based category do not return list with scores, instead the scores serves as the yardstick for ranking. For example, relevance scores are first computed for each document from the emanating source, there after rankings are derived. Note that the ranked ordering can be computed from the relevance scores, but not vice-versa. In environments where document scores are not reported by collections, merging methods assign pseudo-scores to the returned answers[[27] ; [33]]. This survey focuses on the linear score normalization techniques and assumes the availability of relevance scores. There are basically three main linear score normalization techniques: Min-Max [19], Z-Score [23] and Sum [23]. They perform linear transformations of document relevance scores in order to make them comparable across source-specific ranked lists.

Prior study of Result merging has focused primarily on how to get an appreciable number of relevant documents in the final merged results [Callan [1];Callan, Lu [2]; Chakravarthy and Haase [3] ; Craswell, Hawking [4]; Rasolofo, Hawking [27]; Rasolofo, Abbaci [26];Si and Callan [33]] with little or no effort in attaining a diversified unified merged list. This to a large extent limits the capability of a DIR system in satisfying and addressing fully the informational need of a user.

3.1 General Framework To Improve Score Normalization

***Diversified*MinMax**

This algorithm aims at proposing a general framework to score normalization methods in isolation, hence it is assumed that the resource selection phase in DIR has already been completed. For a given user's query q the MinMax method computes comparable relevance scores produced by a particular information source to the range (0..1), by the application of the formula below:

$$(C_{MinMax}(d|q) = \frac{c(d|q) - c_{min}(q)}{c_{max}(q) - c_{min}(q)} \quad (5)$$

where $C(d|q)$ and $C_{MinMax}(d|q)$ are the original and the normalized document scores respectively, *Cmin*(q) and *Cmax* (q) are the minimum and the maximum scores produced by the source for the query q. According to the MinMax technique, the top-ranked document from each information source list, is assigned a normalized score of one. At the merging stage, the ranked lists are combined into a unified list based on the computed normalized document scores. Note that all the top-ranked documents from each source appear on the top of the combined list in a random fashion; because they all have the same normalized score (i.e. one). This score normalization technique and subsequent merging focuses primarily on relevance. For example, normalized documents scores with the same value (e.g. 1) from different rank list are invariably the same. Figure 1.1 below depicts the processes of original MinMax.

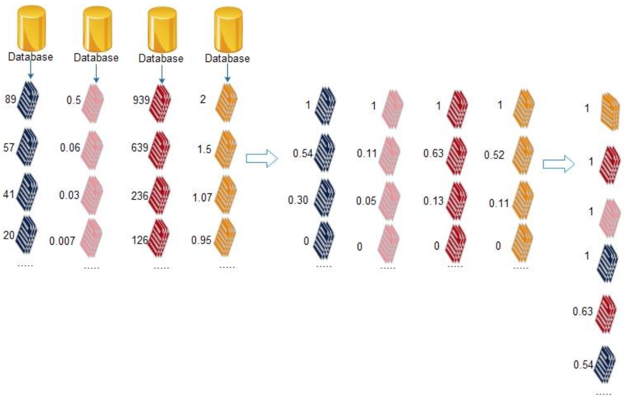

Figure 1.1. The MinMax score normalization. Given a user's query to four different information sources, each produce ranked lists of documents whose relevance scores are not comparable across lists. MinMax normalizes the scores in each list to the range (0..1). Then the documents from all lists are combined and ranked by their normalized scores (with similar score values e.g. 1 , randomly positioned).

Purposed algorithm

The proposed technique aims at placing a retrieved document at a particular rank position during the result merging stage, if this choice satisfies relevance and has a diversity trade off.

Assumptions

We assume that the resource selection phase has already taken place, hence the available selected sources has at least one relevant document for the user query. We also assume the availability of the lexicon statistics of the selected sources(cooperative environment). Lastly we assume the probability relevance scores from each source are assumed to be *data points* hence their *means* and *standard deviations* can be easily computed.

The coefficient of variation (CV)

It is a statistical measure of the dispersion of data points in a data series around the mean. It is calculated as follows:

$$\text{Coefficient of Variation} = \frac{Standard\ Deviation}{Mean} * 100 \quad (6)$$

In the *investing* world, the coefficient of variation allows you to determine how much volatility (risk) you are assuming in comparison to the amount of return you can expect from your investment. In simple language, the lower the ratio of standard deviation to the mean return, the better your risk-return tradeoff.

We argue that randomly placing documents with the same relevance score is one-sided, and would not produce an optimal merged result. Taking a cue from the investing world, as discussed above, it is important to compute the variability of a business (source) to determine the desperation rate in order to make an informed investment decision. Note that early precision is very important in searching, and hence the position of retrieved documents is extremely important especially based on the fact that a user rarely look past the first 10 retrieved documents. For example, assume Col1 , Col2, Col3 … Col50 all have relevant documents, per the MinMax score normalization method discussed, each would have a value of *one* for its top-most ranked document. This means 50 documents would be retrieved first as the most relevant by the user in responds to his query. Arranging these documents in a predefined order, rather than randomly merging them would greatly improve diversity and enhances the user's search experience.

Algorithm 1

1. Given a ranked list of various *P(d/q)* values (denoted by *X*) from each sources Col1 … Coln , their *means* and *standard deviations* can be calculated as follows :

   $$\vec{X}\ (Coli) = \sum_{i=1}^{n} \frac{X(i)}{n} \quad (7)$$

   $$Stand\ Dev(Coli) = \sqrt{\frac{1}{n-1}\sum_{i=1}^{i=n}(X_I - \vec{X})^2} \quad (8)$$

   Where $\vec{X}$ (Coli) and Standard Deviation(Coli) represents the means and standard deviations of data points in each information source respectively and n representing the total data points per each source.

2. Compute the coefficient of variation for each source to determine the rate of variability using the formula below :

   $$\text{Coefficient of variation (CV)} = \frac{\text{Standard deviation}}{\text{Mean}} * 100\% \quad (9)$$

   A typical example of computed mean, standard deviation and CV for figure 1.1 is illustrated in table 1.1 below.

Table 1.1 A theoretical representation of various information sources with individual mean, standard deviations and coefficient of coefficient computed to determine the variability rate in each source.

	Database 1	Database 2	Database 3	Database 4
	89	0.5	939	2
	57	0.06	639	1.5
	41	0.03	236	1.07
	20	0.007	126	0.95
Mean	**51.75**	**0.14925**	**485**	**1.38**
Standard Deviation	**29.09037642**	**0.234838065**	**374.4836445**	**0.476025209**
Coefficient of Variation (CV)	**56.2133 %**	**157.3454%**	**77.2131%**	**34.4946%**

3. Normalize each return relevance score list with the MinMax score normalization technique in (1)
4. Merging is done with emphases on relevance score and diminishing CV value as shown in Figure 1.2

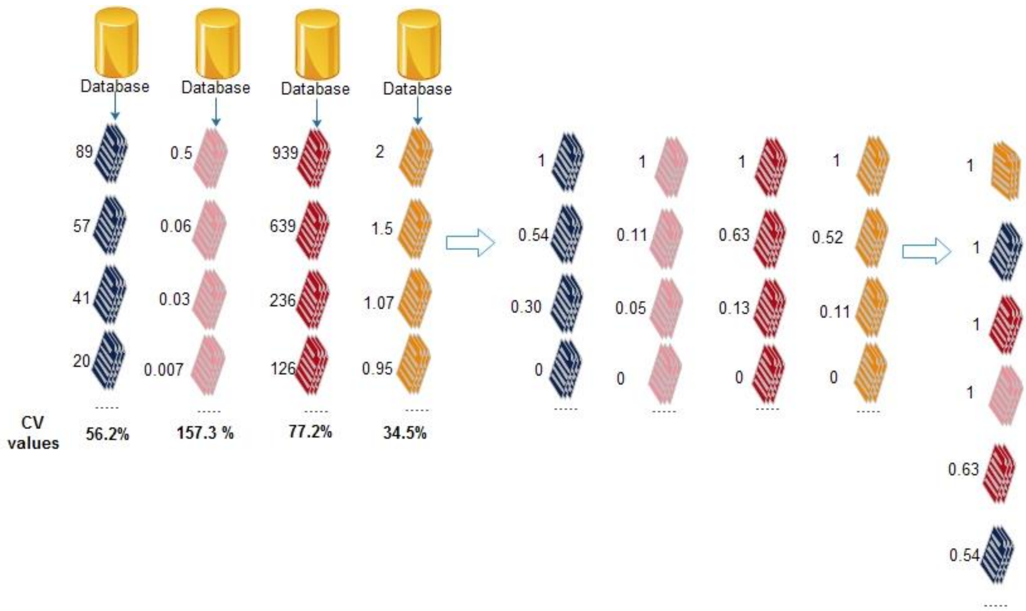

Figure 1.2. The diversified MinMax score normalization. Given a user's query to four different information sources, each produce ranked lists of documents whose relevance scores are not comparable across lists. MinMax normalizes the scores in each list to the range (0..1). Then the documents from all lists are combined and ranked by their normalized scores (with similar score values pruned by CV)

Note: The coefficient of variation is a helpful statistic in comparing the degree of variation from one data series to the other, although the means are considerably different from each other. The higher the coefficient of variation, the greater the level of dispersion around the mean, meaning when the CV for a particular series of data is high it implies the mean is not a good measure of the data points(i.e. the values are very dispersed). The value of CV could have two interpretation in retrieval systems.

A source whose CV is high could imply a few documents are relevant(having high score) with the remaining documents not relevant (having low score) .

A source whose CV is low could imply the differences in the data point scores are marginal or the scores are really similar.

Algorithm 2
***Diversified*Sum**
The original ***sum*** score normalization method proposed by Montague and Aslam [23] first shifts documents scores produced by a particular information source, with the intention of making the minimum score in a ranked list zero - as shown in equation (10).

$$\acute{C}(d|q) = C(d|q) - C_{min}(q) \qquad (10)$$

where *Cmin(q)* is the smallest score attained by a document in a particular information source for the query *q*. The scores are then normalized so that their composite sum is equal to the value one:

$$C_{sum}(d|q) = \frac{\acute{C}\,(d|q)}{\sum_i \acute{C}(d|q)} \qquad (11)$$

$C_{sum}(d|q)$ represents the Sum score normalization technique. For a given user's query, sources produce varied ranked lists of documents modeled with an exponential distribution, which is different across lists. Sum scales these distributions to the same exponential and combine .

With the same intuition derived from ***algorithm 1*** above, we incorporate CV in determining the intra-desperation rate of the documents to influence the merging process.

4 CONCLUSION AND FUTURE DIRECTION

4.1 Conclusions

This survey shows that there is a myriad of approaches that seeks to improve combing results from varied information sources and /or retrieval algorithm. We have analyzed many of these approaches from multiple research domains in the light of score normalization. We have also proposed a general frame to improve results diversification in merging results.

4.2 Future Research

We can expect score normalization to attract research direction for the next decades. In this section, we list some challenging (and promising) research directions.

In most cases, score normalization methods are implemented by merging systems in IR in order to optimize search results. However, the normalization process is considered to be a time consuming task and is sometimes impracticable. It would be important to find other efficient methods that do not use the score normalization process, in order to retrieve Web documents rapidly.

Another issue has to do with diversification during normalization in DIR. Appreciable work has been done in ad hoc search in relation to diversification. Unfortunately apart from the work done by Hong and Si [14] on search result diversification in DIR and another by Ghansah and Benuwa [11] on Fingerprint Based Approach for Resource Selection in Federated search, little is known about diversification in DIR, hence much study is needed in this direction to facilitate the effectiveness of DIR in meeting a searcher's need.

The common aim of merging techniques has been to normalize document and sources scores or use linear regression and curve fitting over the score distribution of sampled documents to compute the final score of a document. The common ignored fact is that precise comparable scores do not automatically optimize precision. Developing merging techniques that can be optimized for different evaluation metrics can be considered as a direction for future investigation.

Again merging becomes more difficult in environments such as aggregated search in which different types of results are blended into a single list, hence developing strategies in addressing the merging problem in aggregated search could be an interesting future direction. Lastly using more robust statistics than max and min in the normalization scheme can help achieve significant improvements in our proposed method hence investigating this hypothesis could be a possible future research area.

REFERENCES

1. Callan, J., *Distributed information retrieval*, in *Advances in information retrieval*. 2000, Springer. p. 127-150.
2. 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.
3. Chakravarthy, A.S. and K.B. Haase. *NetSerf: using semantic knowledge to find Internet information archives*. in *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*. 1995. ACM.
4. Craswell, N., D. Hawking, and P.B. Thistlewaite. *Merging Results From Isolated Search Engines*. in *Australasian Database Conference*. 1999.
5. Croft, W.B., *Combining approaches to information retrieval*, in *Advances in information retrieval*. 2000, Springer. p. 1-36.
6. De, A., E.D. Diaz, and V. Raghavan, *A fuzzy search engine weighted approach to result merging for metasearch*, in *Rough Sets, Fuzzy Sets, Data Mining and Granular Computing*. 2007, Springer. p. 95-102.
7. Deloule, F., P. Lambert, D. Beauchêne, and B. Ionescu. *Data fusion for the management of multimedia documents*. in *Information Fusion, 2007 10th International Conference on*. 2007. IEEE.
8. Dong, X.L. and D. Srivastava. *Big data integration*. in *Data Engineering (ICDE), 2013 IEEE 29th International Conference on*. 2013. IEEE.
9. Farah, M. and D. Vanderpooten, *An outranking approach for information retrieval.* Information Retrieval, 2008. 11(4): p. 315-334.
10. Fox, E.A. and J.A. Shaw, *Combination of multiple searches.* NIST SPECIAL PUBLICATION SP, 1994: p. 243-243.
11. 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.
12. He, C., D. Hong, and L. Si. *A weighted curve fitting method for result merging in federated search*. in *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. 2011. ACM.
13. He, D. and D. Wu. *Toward a robust data fusion for document retrieval*. in *Natural Language Processing and Knowledge Engineering, 2008. NLP-KE'08. International Conference on*. 2008. IEEE.
14. Hong, D. and L. Si. *Search result diversification in resource selection for federated search*. in *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*. 2013. ACM.
15. Howe, A.E. and D. Dreilinger, *SAVVYSEARCH: A metasearch engine that learns which search engines to query.* AI Magazine, 1997. 18(2): p. 19.
16. 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.
17. Khudyak Kozorovitsky, A. and O. Kurland. *Cluster-based fusion of retrieved lists*. in *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. 2011. ACM.
18. Kustra, R. and A. Zagdanski, *Data-fusion in clustering microarray data: Balancing discovery and interpretability.* Computational Biology and Bioinformatics, IEEE/ACM Transactions on, 2010. 7(1): p. 50-63.
19. Lee, J.H. *Analyses of multiple evidence combination*. in *ACM SIGIR Forum*. 1997. ACM.
20. Liang, S. and M. de Rijke, *Burst-aware data fusion for microblog search.* Information Processing & Management, 2015. 51(2): p. 89-113.
21. Liu, K.-L., W. Meng, J. Qiu, C. Yu, V. Raghavan, Z. Wu, Y. Lu, H. He, and H. Zhao. *AllInOneNews: development and evaluation of a large-scale news metasearch engine*. in *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*. 2007. ACM.
22. 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.
23. 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.
24. Paltoglou, G., M. Salampasis, and M. Satratzemi. *Hybrid results merging*. in *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*. 2007. ACM.
25. Qin, T., X. Geng, and T.-Y. Liu. *A new probabilistic model for rank aggregation*. in *Advances in neural information processing systems*. 2010.
26. Rasolofo, Y., F. Abbaci, and J. Savoy. *Approaches to collection selection and results merging for distributed information retrieval*. in *Proceedings of the tenth international conference on Information and knowledge management*. 2001. ACM.
27. Rasolofo, Y., D. Hawking, and J. Savoy, *Result merging strategies for a current news metasearcher.* Information Processing & Management, 2003. 39(4): p. 581-609.
28. Shaw, J.A. and E.A. Fox, *Combination of multiple searches.* NIST SPECIAL PUBLICATION SP, 1995: p. 105-105.
29. Sheldon, D., M. Shokouhi, M. Szummer, and N. Craswell. *LambdaMerge: merging the results of query reformulations*. in *Proceedings of the fourth ACM international conference on Web search and data mining*. 2011. ACM.
30. Shokouhi, M., *Segmentation of search engine results for effective data-fusion*. 2007: Springer.
31. Shokouhi, M. and J. Zobel, *Robust result merging using sample-based score estimates.* ACM Transactions on Information Systems (TOIS), 2009. 27(3): p. 14.
32. 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.
33. 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.
34. Tripathi, A., *Data fusion and matching by maximizing statistical dependencies*. 2011.
35. Tsagkias, M., M. de Rijke, and W. Weerkamp. *Linking online news and social media*. in *Proceedings of the fourth ACM international conference on Web search and data mining*. 2011. ACM.
36. Wu, S., *Fusing Results from Overlapping Databases*, in *Data Fusion in Information Retrieval*. 2012, Springer. p. 149-180.
37. Wu, S., J. Li, X. Zeng, and Y. Bi, *Adaptive data fusion methods in information retrieval.* Journal of the Association for Information Science and Technology, 2014.
38. Wu, S. and S. McClean, *Performance prediction of data fusion for information retrieval.* Information processing & management, 2006. 42(4): p. 899-915.