Applying Data Fusion Methods to Passage Retrieval in QAS

Christensen, Hans Ulrich; Ortiz-Arroyo, Daniel

Published in:
Multiple Classifier Systems

DOI (link to publication from Publisher):
10.1007/978-3-540-72523-7_9

Publication date:
2007

Document Version
Early version, also known as pre-print

Applying Data Fusion Methods to Passage Retrieval in QAS

Hans Ulrich Christensen and Daniel Ortiz-Arroyo

Computer Science Department
Aalborg University Esbjerg
Niels Bohrs Vej 8, 6700 Denmark
huc1405@student.aaue.dk, do@cs.aaue.dk

Abstract. This paper investigates the use of diverse data fusion methods to improve the performance of the passage retrieval component in a question answering system. Our results obtained with 13 data fusion methods and 8 passage retrieval systems show that data fusion techniques are capable of improving the performance of a passage retrieval system by 6.43% and 11.32% in terms of the mean reciprocal rank and coverage measures respectively.

Keywords: Data Fusion, Question Answering Systems, Passage Retrieval.

1 Introduction

A *Question Answering System (QAS)* is one type of information retrieval (IR) system that attempts to find exact answers to user's questions expressed in natural language. In an *Open-Domain Question Answering System (ODQAS)*, questions are not restricted to certain topics and answers have to be found in an unstructured document collection. *Passage Retrieval (PR)*, one component of a QAS, extracts text segments from a group of retrieved documents and ranks these passages in decreasing order of computed likelihood for containing the correct answer. Typically, such text segments are referred to as *candidate passages*.

Data Fusion applied to Information Retrieval (IR Data Fusion) is the intelligent combination of a variety of IR system's opinions on a document's relevance to the user's information need represented in a query. If the combined relevance estimate of a document is more accurate than any of the individual IR system's estimates, performance will be improved. The findings of several recent studies on the application of Data Fusion to IR report improvements in performance [3] [5] [16]. Using Data Fusion techniques, a QAS will be able to provide correct answers to more questions.

This paper is organized as follows. Section 2 briefly describes related work on the application of IR Data Fusion methods to a QAS. Section 3 describes the IR Data Fusion methods investigated. Section 4 investigates which normalization schema performs best for IR Data Fusion applied to PR in a QAS. Section 5 describes the evaluation of the Data Fusion methods and the results achieved. Finally, section 6 presents conclusions and future work.

M. Haindl, J. Kittler, and F. Roli (Eds.): MCS 2007, LNCS 4472, pp. 82–92, 2007.

2 Related Work

Although several studies have investigated the application of Data Fusion to QA systems and in general achieved promising results—e.g. Harabagui et al. [4] report a consistent improvements in terms of precision as high as 20%—only few have investigated the potentially beneficial application of Data Fusion to the task of PR within a QAS. Our contribution in this regard is methodological i.e. we explore experimentally diverse techniques to fuse effectively an ensemble of PR systems to improve the overall performance of a QAS.

The internal Data Fusion experiment carried out by Unsunier et al. reported in [18] achieves a consistent improvement in Coverage@n ranging as high as 119% at Coverage@1. However, the machine learning techniques they employed are based on the learned features of answering passages; hence requiring an extra training step.

As part of their evaluation of a number of PR systems, Tellex et. al. [17] experimentally fuse three PR systems achieving a 6% increase in performance in terms of Mean Reciprocal Rank (MRR). Several other IR studies on the application of Data Fusion for document retrieval (e.g. Lee [9] and Montague [12]) have reported important improvements in performance but on ad-hoc document retrieval systems and not specifically in PR for QAS. Contrarily to these previous approaches, the methodology presented in this paper explores what is the best way to fuse the ranking evaluations produced by diverse PR systems.

3 Descriptions of the Fusion Methods

This section describes with some detail the IR Data Fusion methods we applied to Passage Retrieval. Table 1 shows the IR Data Fusion methods grouped according to Montague's classification scheme [12], which is based on two characteristics of the input: (1) *whether similarity scores are available or ranks only* and (2) *whether training data are available or not*. Typically, training data are judged search results, which are then used for learning IR system specific performance weights. Accordingly, this schema consists of four classes of Fusion methods: *ranks only (RO)*, *relevance scores (RS*), *ranks and training data (R+W)*, and *relevance scores and training data (RS+W)*.

Intuitively, the more information a Fusion method uses for fusing n search results, the better it performs on average. Therefore, we expect an improved performance in RS+W Fusion methods followed by RO+W, RS, and finally RO Fusion method, which we expect to perform worst.

3.1 Data Fusion Methods Based on Ranks Only

Based on Social Choice Theory, Montague and Aslam [13] proposed an adapted version of the Condorcet voting method: the *Condorcet-fuse method*, which they found to be just as effective as the CombMNZ Fusion method. The Condorcet-fuse method is a generalization of the Condorcet election process, where the winner of an election is the candidate that beats or ties with every other candidate in

Availability of relevance scores	*Availability of training data*		
		No training data available	**Training data available**
	Only passage ranks available	Condorcet-Fuse Borda-Fuse Tellex et al. algorithm	Weighted Condorcet-Fuse Weighted Borda-Fuse
	Relevance scores available	CombMNZ MEOWA	Linear Combination Weighted MEOWA

Fig. 1. Classification of the 9 IR Data Fusion methods

a pair-wise comparison, such that the result is a ranked list of documents rather than a single winner. This is accomplished by first executing the Condorcet election process for all pairs of IR systems constructing a graph, where documents are nodes and a directed edge $x \rightarrow y$ means that x has received at least as many votes as y and thus must be ranked higher. Then, traversing all nodes of the graph yields the Fused ranking, where the first node is the highest ranked document. While the Condorcet-fuse voting method has a time complexity of $O(n^2 * k)$, n being the number of documents and k the number of IR systems, Montague and Aslam [13] propose an efficient implementation of Condorcet-fuse using a modified QuickSort method.

In [1], Montague and Aslam also introduce a novel voting Fusion method to the problem of Data Fusion in IR: *Borda-Fuse*, which is an adaptation of the Borda Count election process, where voters give candidates a certain amount of points and the winner is the one who makes more points. Evaluation showed that in two of the five tests using TREC test data, Borda-fuse performed better than the best component IR system in the election [1].

Tellex and colleagues [17] propose a simple yet effective method for fusing the search results of multiple PR in a QAS systems. The method combines a passages rank and a simple vote: the total number of passages retrieved by all component PR systems with a specific document ID into a fused relevance score. Application of the Fusion methods to the 3 best performing passage retrieval methods of the PRISE IR system showed that it was able to improve the Mean Reciprocal Rank by 6% [17]. The method calculates a score for each passage as follows.

$$score(a, r) = \frac{1}{r} + docScore(docID(a, r)) \tag{1}$$

Interestingly, Tellex et al.'s method re-ranks the set of top n candidate passages retrieved by all component PR systems and thus does not fuse identical passages thereby avoiding the difficulty of identifying identical passages.

Based on the observation that frequently when Tellex et al.'s Fusion method boosted low ranked passages, those passages in fact were non-relevant, we

propose in this paper a new Fusion method called *Tellex modified*, where the union of top m passages retrieved by all component PR systems is re-ranked. However, although we restricted Tellex et al.'s Fusion method from including passages at lower ranks, the document counts are still computed for the top 300 passages retrieved by all component PR systems.

3.2 Data Fusion Methods Based on Relevance Scores

Fox and Shaw [6] introduce and evaluate the 6 simple Fusion methods of table 1.

Table 1. The six Fusion Methods introduced by Fox and Shaw (adapted from [6])

CombMAX	$r_f(d_i) = \max_{\forall s_j \in S} \left(r_{s_j}(d_i)\right)$
CombMIN	$r_f(d_i) = \min_{\forall s_j \in S} \left(r_{s_j}(d_i)\right)$
CombSUM	$r_f(d_i) = \sum_{\forall s_j \in S} \left(r_{s_j}(d_i)\right)$
CombANZ	$r_f(d_i) = CombSUM/t$
CombMNZ	$r_f(d_i) = CombSUM * t$
CombMED	$r_f(d_i) = \begin{cases} r_{s_{(n+1)/2}}(d_i) & \text{if } n \text{ is odd} \\ (r_{s_n}(d_i) + r_{s_{(n+1)}}(d_i))/2 & \text{if } n \text{ is even} \end{cases}, r_{s_0}(d_i) \geq \ldots r_{sn}(d_i)$

In table 1, $r_f(d_i)$ is the fused rank of the document d_i, $r_{s_j}(d_i)$ document d_i's rank at the IR system $s_j \in S$, the set of IR systems to be fused, and t the number of IR systems retrieving d_i.

Out of the six "comb"-methods, Fox and Shaw provide evidence that CombSUM followed by CombMNZ performs best. They argue that this is because both methods utilize information about *ranking* (since highly ranked documents are preferred) and *voting* (because documents found by multiple systems are preferred). Later Lee [10] found that CombMNZ in fact performs relatively better. Although these variation in effectiveness may be explained by differences in document collections and query sets, it seems reasonable to assume that in general CombMNZ is superior to CombSUM since more recent Fusion experiments supports this observation.

3.3 Data Fusion Methods Based on Ranks and Training Data

A *weighted Data Fusion method* takes into account a component PR system's ability to provide answering passages. This way, important component PR systems have a greater influence on the Fused relevance score.

As suggested by Aslam and Montague [1], Borda-fuse can be extended to a weighted variant: *Weighted Borda-fuse* by multiplying the points, which a PR system S_i assigns to a candidate passage with a system weight α_i. This is equally true for passages, which a PR system retrieves and those not retrieved, which

are given an average score. This way, the relevance assessments of a PR system considered good at providing candidate passages is preferred. Experimental comparison with CombMNZ Fusion method, where non-optimized weights were used, showed that weighted Condorcet Fuse was able to achieve the same level of performance. Thus, by using improved performance weights, Weighted Borda-fuse has the potential of outperforming CombMNZ [13].

Just like Borda-fuse, Condorcet-fuse can be easily extended to take importance weights into account. In *weighted Condorcet-fuse*, rather than crisp votes, each component PR system gives an importance weighted vote. However, this time the importance weights are used in the binary candidate elections, where the sum of weights rather than votes is compared giving preference to the highest sum. Comparison with the CombMNZ Fusion method showed than weighted Condorcet Fuse performed best on three of four TREC data sets [13].

3.4 Data Fusion Methods Based on Relevance Scores and Training Data

Vogt and Cottrell's *Linear combination* (LC) Data Fusion method combines the relevance scores and training data of two or more component IR systems into a combined relevance score per document [19]. In LC, training data are used for calculating *importance weights* based on standard IR metrics thus reflecting the overall ability of an IR system to provide relevant documents. Given both relevance scores and performance weights the aggregated relevance score is calculated using equation 2.

$$s_{LC}(d) = \sum_{\forall s_i \in S} \alpha_i * s_i(d) \tag{2}$$

where $S_{LC}(d)$ is the fused relevance score assigned to the document d, s_i is the ith PR system's of the set of PR systems to be combined S relevance score, and a_i the importance weight assigned to the ith PR systems. In LC, if an IR system does not retrieve a particular document, then the IR systems is assumed to consider it non-relevant. Accordingly, such a document is assigned a relevance score of 0. Although simple, in various experiments LC has performed well often outperforming the base-line system with a margin of 5% to 10%.

Recently, a number of researchers have investigated the application of the *Ordered-Weighted Averaging (OWA)* class of Fuzzy logic averaging operators to Data Fusion. A desirable property of OWA is that the degree of ANDness can be adjusted by changing the OWA weights.

$$OWA(\boldsymbol{v_i}, \boldsymbol{a_i}) = \sum_{i=1}^{n} v_i * a_i \tag{3}$$

where $\boldsymbol{v}$ is a vector of OWA weights $(v_1, v_2, ...v_n)$ with $\sum v_i = 1$ sorted is descending order and $\boldsymbol{a}$ a vector of Fuzzy satisfaction degrees $(a_1, a_2, ...a_n)$, $a_i \in [0, 1]$. Diaz et al. investigated the use of the OWA class of Fuzzy averaging operator to

ad-hoc IR. They found OWA to outperform Borda-fuse when at least 8 IR systems are combined [3]. In [11] it is reported the use of an adapted version of the weighted maximum entropy OWA (MEOWA) operator as a weighted classifier combination method. It was found that the adapted weighted MEOWA operator improved performance measured by the F_1 measure with 6.51% compared to the best performing classifier [11] in the ensemble.

4 Normalization of Relevance Scores

In general, in Data Fusion applied to IR normalization of relevance scores is necessary since different IR systems use different scales for relevance scores. For example, one IR system might chose to use positive real values and another IR system values in the Unit Interval. Two well-known normalization schemes are shift-scale normalization and shift-sum normalization of equations 4 and 5.

$$score^n_{s_k}(d_i) = \frac{score_{s_k}(d_i) - \min\limits_{j=1}^{m} score_{s_k}(d_j)}{\max\limits_{j=1}^{m} score_{s_k}(d_j) - \min\limits_{j=1}^{m} score_{s_k}(d_j)} \tag{4}$$

$$score^n_{s_k}(d_i) = \frac{score_{s_k}(d_i) - \min\limits_{j=1}^{m} score_{s_k}(d_j)}{\sum\limits_{j=1}^{m} score_{s_k}(d_j) - \min\limits_{j=1}^{m} score_{s_k}(d_j)} \tag{5}$$

where $score_{s_k}(d_i)$ is the unnormalized score of document d_i retrieved by the IR system s_k, $score^n_{s_k}(d_i)$ is the normalized score, and m the number of documents retrieved by the IR system in question.

In order to determine which of the 3 normalization schemes including "no normalization" performs best for Data Fusion applied to PR, we performed an experiment, where the three Data Fusion methods utilizing relevance-scores: CombMNZ, CombSUM and MEOWA are tested with all 238 unique combination of the 8 PR systems: FuzzyPRS [2], JIRS TW and JIRS DM [7], LucenePRS, LucenePRS+FuzzyPRS, Terrier in expC2 [14], and Zettair [1] applied to two QA test sets: CLEF03 and TREC11. Accordingly, $2 * 3 * 238 = 1,428$ runs have to be performed. Performance is measured using the three standard QA metrics: Mean Reciprocal Rank (MRR), Coverage@20, and Redundancy@20.

In summary we found that (1) using no normalization consistently performed worse measured by both Coverage@20 and Redundancy@20 indicating the necessity of using a normalization scheme in PR Data Fusion, and (2) applied to PR Data Fusion, in terms of MRR and Coverage@20 shift-max normalization in general performs better than shift-sum normalization.

[1] Zettair and Lucene are popular open source search engines.

5 Evaluation

Using the methodology employed by Aslam and Montague [1] and Tellex et al. [17], we performed two experiments to investigate the capability of a number of Fusion methods' to improve the performance of the PR module in a QA system to provide an answer to 5 questions. In both experiments we used the same IR Data Fusion methods, component PR systems, QA test data and performance metrics described in subsection 1.

The first experiment, which we denote as *brute-force*, was designed to explore a number of PR systems' ability to improve performance no matter the number and performances of the component PR systems combined. Ideally, a Fusion method will be able to consistently improve the performance of the best performing component PR system. Each of the Fusion methods was tested with all different component PR system subsets of sizes $\{2, 3, 4, 5, 6\}$.

In the second experiment - denoted as *best-to-worst* - we investigated the Fusion methods' ability to improve performance by using knowledge of the past performances of the PR systems. The $i \in \{1, 2, 3, 4, 5, 6\}$ best performing PR systems were combined. Since the top 1 to top 6 PR systems amount to 6 different sets of PR systems, 6 runs with each of the Data Fusion methods had to be performed.

5.1 Methodology

Besides the 9 Data Fusion methods described in section 3, we applied subclass weighting to weighted Condorcet-Fuse, weighted Borda-Fuse, LC and weighted MEOWA (IWMEOWA). Thus, the total number of different Data Fusion methods with which we experimented was 13.

As importance weights we used the performance weights computed using the TREC11 and CLEF03 test data. Evaluation of 7 different performance weights found that max-normalized MRR (nMRR) to perform best.

While the two Fusion experiments were being performed, we found it necessary to exclude Condorcet-fuse with question type weights because it consistently worsened performance.

In our experiments we used the following 8 component PR systems: JIRS [7] using the Distance Model, FuzzyPRS [2], JIRS using the Simple Model, FuzzyPRS, FuzzyPRS+LucenePRS, LucenePRS, Swish-e, Terrier PL2 [14] using In expC2 probabilistic model, and Zettair. A *component PR system* must a) be able to automatically process a question set and b) for each question produce a search result, which contains the information on relevance score, document ID and passage text in the required syntax.

As test data we used TREC12's set of 495 questions and the corpus called AQUAINT consisting of $1,033,461$ documents of English news text and CLEF04's 180 question and the AgenciaEFE corpus of $454,045$ Spanish newswire documents. To answer questions automatically for TREC12 we used

Ken Litkowsky's regular expression patterns of correct answers[2] and for CLEF4 we used the pattern supplied with JIRS[3] The TREC12 question set was reduced to 380, since 115 questions do not have a recognizable pattern.

As evaluation metrics we used MRR, Coverage, and Redundancy. *Mean Reciprocal Rank (MRR)* is defined at the average of the reciprocal value of the first hit to each question within the top 5 candidate passages:

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} RR_i \tag{6}$$

where $RR_i = \frac{1}{r_i}$ if $r_i \leq 5$ or 0 otherwise and Q is the set of questions and r_i the rank of the first answering passage in response to a particular question. As is done in the JIRS system [8], we measure coverage on the first top 20 passages. *Coverage* is defined as the proportion of questions for which an answer can be found within the n top-ranked passages:

$$cov(Q, D, n) \equiv \frac{|\{q \in Q | R_{D,q,n} \cap A_{D,q} \neq \emptyset\}|}{|Q|} \tag{7}$$

being Q the question set, D the passage collection, $A_{D,q}$ the subset of D which contains correct answers for $q \in Q$, $R_{D,q,n}$ the n top ranked passages. *Redundancy* is defined the average number, per question, of the top n passages, which contain a correct answer [15].

$$Redundancy(Q, P, n) \equiv \frac{\sum_{q \in Q} |R_{P,q,n} \cap A_{P,q}|}{|Q|} \tag{8}$$

where Q is the set of questions, $R_{P,q,n}$ the n top ranked passages and $A_{P,q}$ the subset of passages P containing correct answers to $q \in Q$.

5.2 Results

When the Data Fusion methods were applied to PR, using 2 different sets of test data (TREC12 and CLEF04), we were able to improve performance measured by a maximum of 6.43% in terms of MRR and by 11.32% in terms of Coverage@20. These results were obtained by fusing 4 PR system and using our modified version of Tellex et. al.'s [17] Fusion method as shown in table 2.

In general we were not able to improve performance when using Redundancy@20. Somewhat surprisingly, the modified Tellex Fusion method neither requires relevance scores of passages nor importance weights of the PR systems to be fused.

We also obtained the results of the brute-force approach employing both TREC12 and CLEF04 as test data. Comparing the performance results achieved

[2] Ken Litkowsky's patterns are available from the TREC website: `http://trec.nist.gov`.

[3] Patterns of correct answers to CLEF QA test data are available from JIRS' website: `http://jirs.dsic.upv.es/`.

Table 2. The MRR and Coverage@20 of Modified Tellex compared to the 2nd best Fusion methods tested with a) TREC12 (top) and b) CLEF04 (bottom) QA test data

Performance metric	MRR					Coverage@20				
Number of PR4QA systems combined	2	3	4	5	6	2	3	4	5	6
Avg. of best component PR4QA systems	0,2999	0,3093	0,3160	0,3208	0,3242	0,5903	0,6066	0,6168	0,6233	0,6274
Modified Tellex (best)	0,3165	0,3291	0,3363	0,3411	0,3444	0,6420	0,6752	0,6870	0,6937	0,6959
Relative performance in %	**5,54**	**6,41**	**6,43**	**6,33**	**6,25**	**8,77**	**11,32**	**11,39**	**11,30**	**10,91**
LC	0,2999	0,3159	0,3269	0,3322	0,3353	0,6130	0,6357	0,6508	0,6586	0,6644
Relative performance in %	**0,00**	**2,15**	**3,46**	**3,55**	**3,43**	**3,85**	**4,79**	**5,51**	**5,66**	**5,89**

Performance metric	MRR					Coverage@20				
Number of PR4QA systems combined	2	3	4	5	6	2	3	4	5	6
Avg. of best component PR4QA systems	0,3522	0,3620	0,3691	0,3745	0,3787	0,5903	0,6066	0,6168	0,6233	0,6274
Modified Tellex (best)	0,3567	0,3666	0,3712	0,3759	0,3785	0,6224	0,6575	0,6727	0,6806	0,6845
Relative performance in %	**1,28**	**1,26**	**0,59**	**0,39**	**-0,03**	**5,45**	**8,40**	**9,07**	**9,19**	**9,10**
LC w. question class weights	0,3497	0,3616	0,3695	0,3765	0,3847					
LC						0,6153	0,6420	0,6547	0,6639	0,6708
Relative performance in %	**-0,70**	**-0,11**	**0,12**	**0,54**	**1,60**	**4,24**	**5,83**	**6,15**	**6,52**	**6,92**

in terms of MRR and Coverage@20 metrics when applying the Fusion methods to both TREC12 and CLEF04 test data revealed that in general only the modified version of Tellex et al.'s [17] method was able to improve performance in terms of these metrics.

For each of the 4 classes of Fusion methods we calculated the average and maximum of the results achieved when all Fusion methods were applied to both test sets. We found that the class of Fusion methods, which only takes a ranked list of candidate passages as input, performed best, contrarily to our expectation that Fusion methods utilizing more information on relevance perform better.

We used a historical MRR metric computed using CLEF03 and TREC11 sets of test data to rank the individual PR systems according to their performances, since we found this metric to perform best. We found that all three performance metrics were consistently either improved or remained the same compared to the results of the brute-force experiments for both TREC12 and CLEF04 test data. Although the results did not reveal a Fusion method, which consistently benefited the most from fusing the i best performing PR systems, the results indicated that the combination of the top 2 PR was able to achieve the highest improvements.

We tested both the unweighted and weighted versions of Condorcet-fuse yielding a total of 4 Fusion methods. While both Fusion methods consistently improved performance for the TREC11 QA test set by a small margin, they fail doing so when applied to CLEF04 QA test data, where performance measured as MRR degrades by a maximum of 3.56% thus indicating a need for more advanced weighting schemes i.e. applying machine learning techniques.

Finally, we tested IWMEOWA, Condorcet-fuse and Linear Combination with both types of weighting resulting in 6 different Fusion methods applied to both TREC12 and CLEF04 test data. We found that using weights per question type instead of overall importance weights did not consistently provide additional performance improvements.

6 Conclusions and Future work

This paper investigated the application of a total of 13 Data Fusion methods to Passage Retrieval in a QAS, eight of these utilize importance weights and importance weight per subclass of questions. Using 2 sets of QA test data, we found that the data fusion mechanisms were able to improve MRR by a maximum of 6.43% and Coverage@20 by 11.32%. These results were obtained by fusing 4 PR in a QAS systems with our proposed modification to Tellex et. al.'s method [17]. Contrarily to our initial expectations, based on the performance improvements obtained by the use of importance weights in voting systems applied in information retrieval systems and [3] patent classification [11], our experiments indicate that importance weights did not yield significant improvements in performance. The reason for this may be that we used a too simplistic approach to obtain the performance weights employed in our experiments and that the proposed question classification scheme does not generalize well with new questions.

As future work we plan to apply machine learning techniques and advanced question classification schemes to Data Fusion of PR systems in a QAS to improve performance even further.

References

1. J. Aslam and M. Montague. Models for metasearch. *The 24th Annual ACM Conference on Research and Development in Information Retrieval (SIGIR 01), New Orleans, LA*, 2001.
2. H. U. Christensen. Exploring the use of fuzzy logic in passage retrieval for question answering, March 2006. Preliminary master's thesis report available online: `http://hufnc.1go.dk/MidtermReport.pdf`.
3. E. D. Diaz, A. De, and V. Raghavan. A comprehensive owa-based framework for result merging in metasearch. *Proceedings of Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing: 10th International Conference, RSFDGrC 2005, Regina, Canada*, 2005.
4. Harabagiu et al. Employing two question answering systems in trec-2005. *In proceedings of The Fourteenth Text REtrieval Conference (TREC 2005)*, 2005.
5. W. Fan, M. Gordon, and P. Pathak. On linear mixture of expert approaches to information retrieval. *Decision Support Systems*, 42:975–987, 2006.
6. E. A. Fox and J. A. Shaw. Combination of multiple searches. *In The Second Text REtrieval Conference (TREC-2), Gaithersburg, MD, USA*, pages 243–249, March 1994.
7. J. Gómez-Soriano and M Montes y Gómez. Jirs—the mother of all the passage retrieval systems for multilingual question answering? `http://www.dsic.upv.es/workshops/euindia05/slides/jgomez.pdf`.
8. J. Gómez-Soriano, M. Montes y Gómez, E. Arnal, L. Villase nor Pineda, and P. Rosso. Language independent passage retrieval for question answering. *Fourth Mexican International Conference on Artificial Intelligence MICAI 2005*, Lecture Notes in Artificial Intelligence, Springer-Verlag, November 2005. Monterrey, Mexico.
9. J. H. Lee. Combining multiple evidence from different properties of weighting schemes. *In proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, July 1995.

10. J. H. Lee. Analyses of multiple evidence combination. *In: Belkin NJ, Narasimhalu AD and Willett P, eds. SIGIR 97: Proceedings of the Twentieth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pages 267–276, July 1997.
11. H. Mathiassen and D. Ortiz-Arroyo. Automatic classification of patents using classifier combinations. In *Proceedings of IDEAL 2006, 7th International Conference on Intelligent Data Engineering and Automated Learning LNCS Vol. 4224*, pages 1039–1047, September 20-23rd 2006.
12. M. Montague. *Metasearch: Data fusion for document retrieval*. PhD thesis, Dartmouth College, 2002.
13. M. Montague and J. Aslam. Condorcet fusion for improved retrieval. *The 11th Annual ACM Conference on Information and Knowledge Management (CIKM 02), Tysons Corner, VA.*, July 2002.
14. I. Ounis, G. Amati, V. Plachouras, B. He, C. Macdonald, and D. Johnson. Terrier information retrieval platform. In *Proceedings of the 27th European Conference on IR Research (ECIR 2005)*, volume 3408 of Lecture Notes in Computer Science, pages 517–519. Springer, 2005.
15. I. Roberts and R. Gaizauskas. Evaluating passage retrieval approaches for question answering. *In Advances in Information Retrieval: Proceedings of the 26th European Conference on Information Retrieval (ECIR04), number 2997 in LNCS*, Lecture Notes in Computer Science, Springer-Verlag:72–84, 2004.
16. A. Spoerri. How the overlap between the search results of different retrieval systems correlates with document relevance. *In Grove, Andrew, Eds. Proceedings 68th Annual Meeting of the American Society for Information Science and Technology (ASIST) 42*, 2005.
17. S. Tellex, B. Katz, J. Lin, G. Marton, and A. Fernandes. Quantitative evaluation of passage retrieval algorithms for question answering. *In Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2003)*, July 2003.
18. N. Unsunier, M. Amini, and P. Gallinari. Boosting weak ranking functions to enhance passage retrieval for question answering. *In IR4QA workshop of SIGIR 2004*, 2004.
19. C. C. Vogt and G. W. Cottrell. Fusion via a linear combination of scores. *Information Retrieval*, 1(1):151–173, October 1999.