

Toward a Robust Data Fusion for Document Retrieval

Daqing He

School of Information Sciences
University of Pittsburgh
Pittsburgh, PA, USA
email: dah44@pitt.edu

Dan Wu

School of Information Management
Wuhan Univeristy
Wuhan, HuiBei, China
Email: woodan@whu.edu.cn

Abstract

This paper describes an investigation of signal boosting techniques for post-search data fusion, where the quality of the retrieval results involved in fusion may be low or diverse. The effectiveness of data fusion techniques in such situation depends on the ability of the fusion techniques to be able to boost the signals from relevant documents and reduce the effect of noise that often comes from low quality retrieval results. Our studies on Malach spoken document collection and HARD collection have demonstrated that CombMNZ, the most widely used data fusion method, does not have such ability. We, therefore, developed two versions of signal boosting mechanisms on top of CombMNZ, which result in two new fusion methods called WCombMNZ and WCombMWW. To examine the effectiveness of the two new methods, we conducted experiments on Malach and HARD document collections. Our results show that the new methods can significantly outperform CombMNZ in combining retrieval results that are low and diverse. When the tasks are to combine retrieval results that are in similar quality, which have been the scenarios that CombMNZ are applied often, the two new methods still can obtain often better, sometimes significantly, fusion results.

Keywords: Data fusion, Spoken document retrieval, CombMNZ, Malach, TREC HARD.

1 Introduction

With the amount of information in various formats or media available on the Web, the search for relevant documents for a given topic often can rely on information from multiple sources to improve the results. There could be three possible scenarios of integrating multiple sources of information. First, the combination can happen at the query side, where query terms from multiple sources are combined to generate a single query to be entered to a retrieval system (e.g., Belkin and his colleagues' study on progres-

sively combining Boolean query formulations from multiple users [2]). Another school of studies that belongs to this scenario is query expansion by relevance feedback [6, 12]. The various sources explored in TREC Robust and HARD tracks to obtain further information about retrieval topics are also the examples [1, 16]. The second scenario of combining multiple information happens at the result side. Search results related to one topic (query) from different search engines are combined to generate a single outcome. What happens inside Web meta-search engines belongs to this scenario [9]. The third scenario concerns the situation that there could be multiple representations of the same document, and the integration happens at the result side to merge outcomes of the searches on each representation into one single result. One example of this scenario is the retrieval performed on the MALACH test collection, which contains manually constructed segments from 300 interviews of Holocaust survivors. Each segment contains two versions of automatic transcriptions, two sets of automatically generated thesaurus terms, and manually generated summaries, thesaurus terms and person names. Each of them can be viewed as the representation of the documents.

In the literature, the techniques for combining multiple queries, document representations or retrieval results are called "data fusion" [7]. Belkin et al. [2] studied progressive pre-search data-fusion from multiple users. Their results showed that combining queries led to progressively improving retrieval performance. Lee [7] analyzed post-search data fusion methods using TREC3 data, and identified the rationale for combining different results as that different runs retrieve similar sets of relevant documents, but retrieve different sets of irrelevant documents. Several studies have also been concentrated on applying Dempster-Shafer theory for combining evidence [14]. The theory provides a principle way of combining belief of the data into fusion calculation, but the fusion process of three or more results has to be aggregated through a set of combination of two results, which could be cumbersome.

We believe that the data fusion method should be able

Table 1. Retrieval results on MALACH Spoken Document Collection

runs	MAP	runs	MAP
asr04	0.0693	manual	0.2312
autowa1	0.0464	altogether	0.1842

to handle more than two sources at the same time, and it should be able to not only combine results that are with reasonable retrieval effectiveness, but also fuse results that are low in performance. For example, as show in Table 1, there is great difference in effectiveness between the retrieval run “manual”, which is based on manual only information (i.e., on manually generated summaries, thesaurus terms, and person names), and the run “asr04”, which is based on the automatic speech recognition (ASR) outputs. However, close examination of the return documents shows that “asr04” runs picked up many documents that were missed by the “manual” run. In addition, the simple combination approach that searches all the information in a document together (i.e., the “altogether” run) does not solve the problem as the “altogether” run is significantly inferior than the “manual” run. Therefore, the best strategy in this type of situations is not abandon the bad performance runs totally, or using a simple combination method, but to develop techniques to extract the useful information from the bad runs.

Data fusion with low performance runs can happen in the second scenario mentioned above too. The Web logs analysis conducted by Spink et al [13] shows that the overlap between Web search engines is relative low, which makes meta-search engine useful in finding all relevant information for a given topic. Since not all search engines perform equally well for a given topic, the data fusion algorithms have to combine results from bad performance results to obtain high recall in the results, but at the same time reduce the noise from the bad performance runs to maintain the quality of the overall results.

In this paper, we will study data fusion techniques for extracting useful information from bad performance retrieval results, in particular, we will concentrate on the two situations: 1) data fusion is performed on results from multiple search engines, and 2) data fusion is performed on the results from multiple representations of a document. Here we assume that the retrieval results are in the form of ranked lists, and we are interested in combining results from three or more ranked lists. For all the above reasons, we select CombMNZ, a data fusion algorithm for combining multiple ranked lists, as the method for our study. This algorithm was first raised by Fox and Shaw [5], and later studied by Lee [7] as the best performed algorithms among commonly used. Our research questions are presented below, and we intend to provide answers to them via experimental studies.

1. Is data fusion algorithm CombMNZ capable of handling retrieval results with diverse effectiveness under the two above situations?
2. If it cannot, what could be the modifications in the algorithm to resolve this problem?

In the remaining of this paper, we will first discuss the data fusion algorithm CombMNZ in detail in Section 2. Then in Section 6, we will talk about our experiments for exploring the answers to the two above research questions, which end up revealing the revised method for combining diverse runs. We then end with a review of the related work in section 7, and a conclusion of the major contributions and future work in section 8.

2 Post-Search Data Fusion

Data fusion has been an active topic in text retrieval process, and people have developed many techniques for applying fusion techniques in various retrieval applications. In post-search data fusion approaches, two major issues are 1) how to normalize the scores of different runs so that the scores are comparable; and 2) how to combine the normalized scores to obtain a new and hopefully better ranked list.

Score normalization is important to cope with the situation where one set of scores can be probabilities between 0 to 1, whereas another set can be the logarithm numbers between 0 to negative infinity. A often used normalization method (see Equation (1)) utilizes the maximum and minimum scores of a ranked list (i.e., *MaxScore* and *MinScore*) in the normalization process [7].

$$NormedScore = \frac{OrgScore - MinScore}{MaxScore - MinScore} \quad (1)$$

Fox and Shaw [5] proposed several fusion methods for combining multiple scores (see Table 2). Some methods select one extreme end of the sample values to be the representative score of a document in the fused results (e.g., CombMIN and CombMAX), whereas the others use some form of the sum of all sample values as the final score (e.g., CombSUM, CombANZ, CombMNZ). Some methods also emphases or de-emphases those documents that appear multiple times in the different results (e.g., CombANZ and CombMNZ). Lee [7] conducted experimental studies on these methods, and established that CombMNZ is the best among the five methods in retrieving TREC ad hoc data. This is why we will concentrate on examining the CombMNZ algorithm in our studies.

Table 2. Combining functions by Fox and Shaw

CombMIN	minimum of all scores of a document
CombMAX	maximum of all scores of a document
CombSUM	summation of all scores of a document
CombANZ	$\text{CombSUM} \div \# \text{ of nonzero scores of a document}$
CombMNZ	$\text{CombSUM} \times \# \text{ of nonzero scores of a document}$

3 Experiment Settings

3.1 Measure

As stated, the data fusion we are studying is performed on multiple ranked lists, whose result is also a ranked list. The most frequently used summary measure for a ranked list is Mean Average Precision (MAP), which aims at giving an overview of the quality of a ranked retrieval run with both precision and recall oriented emphasis. Because it is sensitive to the entire ranking, we use MAP as the measure to examine closely the quality of the fusion results.

3.2 Data

To examine data fusion techniques for both the second and third scenario (see Section 1), our studies consisted of experiments on retrieval results from two different collections of documents. The first set corresponded to the third scenario where results from multiple representations of a documents are to be combined. It had a group of ranked lists from our participation to spoken document retrieval track of CLEF 2005 (see Table 1). The data collection used in those experiments was MALACH Test Collection, which was developed by University of Maryland as part of their effort in MALACH project [8]. The collection contains about 7800 manually constructed segments from 300 interviews of Holocaust survivors. Every segment has the following:

- two automatic speech recognition outputs from the ASR system developed by IBM in 2003 and 2004 respectively. The word error rate (WER) of the two outputs are about 40% and 35% respectively.
- two sets of automatically generated thesaurus terms by University of Maryland;
- a set of human generated data, including person names mentioned in the segment, average 5 thesaurus labels and 3-sentence summaries.

This collection has a total 63 search topics in TREC style with a title, a description and a narrative. The topics were available in multiple languages, but we only used the English version for our studies. All the retrieval results were

generated by Indri 1.0 search engine ¹. In the remaining of this paper, we will call this set of results *MALACH runs*.

The other set of results were used to simulate the second scenario where results from multiple different search engines are to be combined (see Table 3). They were from High Accuracy Retrieval from Documents (HARD), a track in TREC 2005. The data collection was AQUINT corpus with 1,033,461 news articles from multiple news agencies and 50 topics from previous year’s Robust track. The results used in this study came from two sites: University of Pittsburgh and University of Massachusetts Amherst. The retrieval systems used in both universities happened to be Indri 2.0 search engine, but the pre-processing of documents, the building index of the collections, and the actually retrieval setting of the Indri system were done independently at the two sites. Therefore, it can be seen that the results were generated from two different search engines. In this paper, we call this set *HARD runs*.

Besides the benefits of examining the data fusion techniques in more diverse environment, another benefit of working on both MALACH runs and HARD runs is that we can examine the data fusion techniques in both new genre environment (spoken document retrieval) and more traditional TREC environment (HARD).

Table 3. Retrieval Effectiveness of HARD runs

runs	MAP	runs	MAP
PPPP1	0.2566	MMMM1	0.0526
PPPP3	0.2908	MMMM2	0.0814
PPPP4	0.2637	MMMM3	0.3039
		MMMM6	0.3019

3.3 Baselines

When examining the first research question about the effectiveness of using CombMnZ for data fusion of diverse retrieval results, we use the best single run that participates the data fusion as the baseline. The purpose is to examine whether the fusion achieves significant changes by comparing to the best we can get without the fusion.

When examining the revisions on CombMNZ, besides employing the best single run mentioned above, we will use CombMNZ as the other baseline so that we know whether the revisions make any difference in data fusion.

¹<http://newhaven.lti.cs.cmu.edu/indri/>.

4 CombMNZ in Combining Diverse Ranked Lists

Lee’s studies show that CombMNZ can achieve better results by combining multiple ranked lists that are in comparable performance [7]. However, we do not know how close the performance between the ranked lists should be to qualify as comparable performance. What would happen if one or several ranked lists are not comparable?

To identify possible answers to these questions, we designed data fusion experiments involving ranked lists with multiple degree of diversity in performance. Using the HARD runs as the example, the data fusion tasks are:

- LL (i.e., LowLow run): MMMM1 + MMMM2; This fusion task is performed on two ranked lists with low MAP values. Although the absolute difference between both values is only about 0.03, in relative sense, MMMM2 is about 60% increase to MMMM1. Therefore, this fusion task represents the situation for combining ranked lists with low and diverse performance. The task ID is HDMNZLL.
- LLH (i.e., LowLowHigh run): MMMM1 + MMMM2 + PPPP1; This fusion task represents the combination of a much better ranked list with two really low ranked lists. Now the difference between MMMM1 and MMMM2 is much smaller than the difference between either of them to PPPP1. The task ID is HDMNZLLH.
- LHH: MMMM2 + PPPP1 + PPPP3; This fusion task contains two relative competitive ranked lists PPPP1 and PPPP3, and a much lower ranked list MMMM2. The task ID is HDMNZLHH.
- HHH1: PPPP1 + PPPP3 + PPPP4; This task contains three relative competitive ranked lists, which is a situation close to the tests in Lee’s study. The task ID is HDMNZHHH1.
- HHH2: PPPP3 + MMMM3 + MMMM6. This fusion task is similar to HHH1, but the three runs are even better. The task ID is HDMNZHHH2.

As shown in Table 4, the fusion results obtained using CombMNZ method did affected by the quality of the data that it tried to combine. When the ranked lists are reasonable close to those studied by Lee, the fused runs did performed better or close to the best single baseline run (see the row for HDMNZHHH2), however there were situations that some fusion results were actually inferior results to the baseline (see the row for HDMNZHHH1). When the ranked lists are diverse, as shown in top three rows in Table 4, CombMNZ not only could not extract useful information

from low perform ranked lists, but also it affected by the noise from low quality ranked lists which end up generated much inferior, and often significantly inferior, results than the corresponding best single run. It seems that CombMNZ does not have the right mechanism to cope with much noise existing in low performance ranked lists.

Table 4. Data fusion experiments using CombMNZ on HARD runs. The percentages in bold indicate a statistically significant difference by paired t-test with $p \leq 0.05$

Fused runs	MAP	Best single run	MAP	relative MAP change
HDMNZLL	0.0693	MMMM2	0.0814	-15%
HDMNZLLH	0.1625	PPPP1	0.2566	-26%
HDMNZLHH	0.2359	PPPP3	0.2908	-8%
HDMNZHHH1	0.2805	PPPP3	0.2908	-4%
HDMNZHHH2	0.3337	MMMM6	0.3019	10%

We performed similar data fusion tasks on MALACH runs. Due to the availability of the retrieval results, we were only able to perform three fusion tasks:

- MCMNZLL: asr04 + autowa1;
- MCMNZLLH: asr04 + autowa1 + manual;
- MCMNZLHH: asr04 + manual + altogether;

Table 5. Data fusion experiments using CombMNZ on MALACH runs. The percentages in bold indicate a statistically significant difference by paired t-test with $p \leq 0.05$.

Fused runs	MAP	Best single run	MAP	Relative MAP change
MCMNZLL	0.0759	asr04	0.0693	9.52%
MCMNZLLH	0.1597	manual	0.2312	-30.93%
MCMNZLHH	0.2121	manual	0.2312	-8.26%

As shown in Table 5, except in LL task, similar results were obtained in data fusion on MALACH runs on LLH and LHH tasks. CombMNZ method was sensitive to the quality of the ranked lists participating in the data fusion process. The diverse the lists are, the less chance it could produce better fusion results than the baseline. This essentially defeats the purpose of using data fusion to those ranked lists. However, it is interesting to see that CombMNZ achieved better fusion results over the baseline on LL task, although the difference is not significant.

5 Signal Boosting in CombMNZ

Close examination on CombMNZ method reviews two assumptions employed in the method: 1) when calculating the sum of the normalized score of a document, it assumes that different ranked lists have equal contribution. The only difference is from the normalized score of that document in each ranked list; 2) when promoting potential relevant documents over irrelevant documents by examining the number of times that a document appearing in multiple ranked list, it assumes that it is equally important to know as an evidence that a document appears in a ranked list regardless the quality of the ranked list.

A ranked list can be viewed as a set of documents recommended by the retrieval system to be the potential relevant documents. The score or the rank associated with each document can be viewed as the belief that the system assigned to the document for its potential relevance. When a ranked list has many truly relevant documents with high ranked scores, the recommendation from the ranked list are often accurate. When a ranked list contains many irrelevant documents at the top, its recommendations are often false. For example, we would obtain more relevant documents from a ranked list containing 60 truly relevant documents out of 100 returned documents than from a ranked list containing just 6 truly relevant document out of the same number of returned documents. Therefore, intuitively, we would trust more on the recommendations from the first list than that from the second. This situation is similar to assigning different belief to result sets in Dempster-Shafter theory [11].

The score normalization process, which is necessary in data fusion techniques like CombMNZ, makes thing even worse. It removes any possibility to use the characteristics of original scores in the ranked list to catch some cues about the quality difference among the ranked lists.

Therefore, because of the build-in assumptions in the CombMNZ method, it is more suitable for combining ranked lists with relative similar performance. If we want to perform data fusion on ranked lists with diverse performance, we have to be able to specify certain belief about the quality of the recommendations from individual ranked lists. If we model an accurate recommendation about the relevance of a document as a signal, and a false recommendation of such as a noise, the mechanism that should be built in the data fusion techniques should have some signal boosting mechanism to promote signals over noises.

We have established that the reason that CombMNZ method does not perform well with diverse results is because it cannot distinguish well the evidence from high quality results, which has higher probability to be signals, from that from low quality results, which has high probability to be noises. Therefore, one revision is to boost the potential signals from high quality results, and reduce the

effect of the potential noises from low quality results.

Consequently, we proposed two possible revisions to CombMNZ to accommodate the different belief of the recommendation quality, which results in two algorithms for data fusion called WCombMNZ and WCombMWW.

Both WCombMNZ and WCombMWW depend on WCombSum, which combines the normalized scores of a document with a predefined weight assigned to that particular ranked list (see Equation 2).

$$WCombSUM_i = \sum_{j=1}^{m_i} (w_j \times NormalizedScore_{i,j}) \quad (2)$$

where $WCombSUM_i$ is the final score for a document i , w_j is a predefined weight associated with ranked list j , m_i is the number of nonzero scores of document i , and the $NormalizedScore_{i,j}$ is calculated using Equation 1.

WCombMNZ still assumes the equal evidence about the appearance of a document in the ranked lists, so the final score of a document in fusion ranked list is

$$WCombMNZ_i = WCombSUM_i \times m_i \quad (3)$$

where m_i is the number of nonzero scores of document i .

On the contrary, WCombMWW assumes that there is difference between accepting the evidence of a document appearing in a high quality ranked list and accepting that in a low quality list. Therefore, using a weight with a ranked list mentioned in WCombSUM, The final score of a document i in WCombMWW is defined as in Equation 4.

$$WCombMWW_i = WCombSUM_i \times K \sum_{p=1}^{m_i} w_p \quad (4)$$

where m_i is the number of nonzero scores for document i , w_p is the weight for a ranked list, which is the same as w_j in Equation 2, and K is a predefined factor for a give set of ranked lists. Since K is applied to every document from all the ranked lists, its value in principle does not affect the results of the data fusion at all. It is added into the equation for a practical reason since our scripts for handling the data fusion cannot handle well the numbers that are smaller than 10^{-4} . In our studies, K is always assigned as 1000.

Many schemas can be developed to assign weights w_j to a ranked list j . In our studies, we used the MAP value of a retrieval system on some training topics as the as W_j of the ranked list. These MAP values can be seen as the surrogates or the belief assigned to the specific system/setting based on previous knowledge. Since all the retrieval topics we used were developed independently, this arrangement makes sense and resembles the real situation when people have developed certain belief about the quality of specific system after interacting with it for a while.

6 Experiments

6.1 Experiments on MALACH runs

To conduct experiments on examining the usefulness of WCombMNZ and WCombMWW, we divided the topics associated with both MALACH runs and HARD runs into training parts and testing parts. The training parts were used for obtaining possible optimal combinations of weights w_j for a given set of ranked lists. Among the 63 topics associated with MALACH runs, we divided the first 30 topics for training, and the remaining 33 topics for testing²

Table 6. The ranked lists of MALACH runs in training and testing parts

Training runs	MAP	Testing runs	MAP
manual	0.2279	manual	0.2340
autowa1	0.0230	autowa1	0.0660
asr04	0.0573	asr04	0.0796
alltogether	0.1876	alltogether	0.1813
MCMNZLL	0.0529	MCMNZLL	0.0955
MCMNZLLH	0.1444	MCMNZLLH	0.1728
MCMNZLHH	0.2177	MCMNZLHH	0.2073

Due to lack of means to explore the space of all possible combinations of weights, we simplified the weight training process to be the increasing of the weight of the best ranked list in data fusion by multiple times of its MAP score. The weights of other ranked lists will always be their MAP values. The rationale of this approach is that we believe that the signal boosting can be achieved by enhancing the strength from the best ranked list.

The experiments for testing WCombMNZ and WCombMWW on MALACH runs were performed on LL, LLH and LHH three fusion tasks. The ranked list involved in these three tasks were the same as the ones used in our study of CombMNZ (see Section 4). As shown in Table 7, using MAP values of the ranked lists as their weights gave us a reasonable starting point. Although the fusion lists obtained this way did not always give us the best results in the training, the difference was reasonable small except in WMNZLLH. At least in these tasks on MALACH runs, the training seems not to be really critical.

As shown in Figure 1, both WCombMNZ and WCombMWW performed well at combining ranked lists with low or diverse results. Comparing to the CombMNZ

²We acknowledge that this division of the topics is ad hoc. We will try various separations of the topics in the future to remove the possible influence from individual topics. Same for the HARD topics.

Table 7. Experiments for WCombMNZ and WCombMWW on MALACH runs.

Type	Fused runs	MAP
Training best	MCWMNZLL-2map	0.0567
Testing	MCWMNZLL-map	0.0934
Training best	MCWMWVLL-2map	0.0600
Testing	MCWMWVLL-map	0.0944
Training best	MCWMNZLLH-5map	0.2090
Testing	MCWMNZLLH-5map	0.2185
Training best	MCWMWVLLH-3map	0.2410
Testing	MCWMWVLLH-3map	0.2421
Training best	MCWMNZLHH-2map	0.2361
Testing	MCWMNZLHH-2map	0.2245
Training best	MCWMWVLLH-2map	0.2449
Testing	MCWMWVLLH-2map	0.2332

baselines, the two new methods achieved significantly better results in LLH and LHH tasks (measured by paired t test ($p \leq 0.05$)). The relative improvement ranged from 8% at minimum to 40% at maximum. The two news methods did perform inferior on LL tasks, but the difference was not significant. Comparing to the other baseline, the best pre-fusion run in the ranked lists, our two new methods achieved near 20% relative improvement on MAP on LL task. However, the difference was not significant. In other fusion tasks (i.e., LLH and LHH), the two new method only achieved comparable results to the single best run baseline. Between the two new methods, WCombMWW constantly outperformed WCombMNZ. Overall, it seems that, as we hoped, the introduction of signal boosting techniques made the data fusion process more robust to the noise associated with low performance ranked lists.

6.2 Experiments on HARD runs

We also divided the 50 topics for HARD runs into 25 training topics and 25 testing topics. Same to our examination of CombMNZ in HARD runs, the fusion tasks include LL, LLH, LHH, HHH1, and HHH2 (see Section 4 for the details of these fusion tasks).

Comparing to the results obtain from fusions on MALACH runs, Figure 2 shows similar patterns on LLH and LHH tasks. In both tasks, the two new methods significantly outperformed CombMNZ, and the two new methods failed to achieve significant improvement over the other baseline, the best pre-fusion run in the ranked lists. In addition, although the two new methods obtained near 20% relative improvement over CombMNZ on LL task, same as its improvements over the other baseline, they are not statistically significant. However, the perfor-

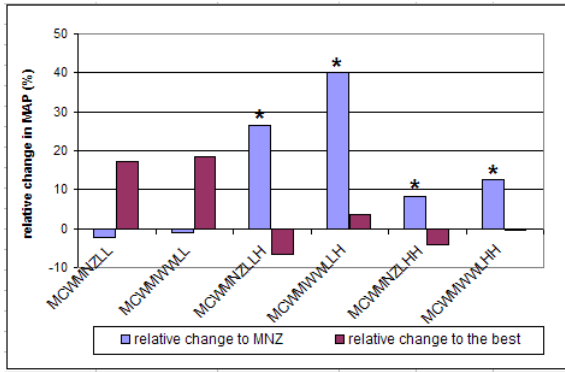


Figure 1. Relative MAP changes on MALACH runs between WCombMNZ, WCombMWW and the two baselines: corresponding CombMNZ runs and the best pre-fusion runs in the fusion process. All the runs related WCombMNZ contain “WMNZ” in their names, and those related to WCombMWW contain “WMWW”. The “*” on the top of a bar indicates that the change is statistically significant measured by paired t-test.

mance of the two new methods did achieve significant improvement in HHH2 task over the single best run baseline, and WCombMWW achieved significant improvement over the CombMNZ in HHH1 task. Here, the superiority of WCombMWW over WCombMNZ is not as clear as in experiments on MALACH runs. Although the former achieved much better results in LLH task, its performance in other tasks was slightly worse than that of WcombMNZ.

7 Related work

Data fusion has been an active research area. Turtle and Croft developed an inference network that can combine different documents representations and different versions of a query in a probabilistic framework [15]. Belkin et al [2] studied the effect of progressively combining Boolean query formulations from multiple users. Their results showed that progressively combining queries led to progressively improving retrieval performance. Fox & Shaw [5] proposed a set of post-search data fusion methods, which were used by Lee [7] to study the effectiveness difference among the methods. Lee’s study demonstrated that CombMNZ is the most effective method on TREC3 ad hoc retrieval data. Lee also stated that the reason that data fusion techniques work is because different retrieval results would have greater overlap of relevant documents than that of irrelevant documents. Several studies have also been concentrated on applying Dempster-Shafer theory for combining evidence [14, 11]. The theory provides a principle way of combining belief of the data into fusion calculation, but the fusion process of three or more results has to

Table 8. The ranked lists of HARD runs in training and testing parts

Training runs	MAP	Testing runs	MAP
PPPP1	0.2450	PPPP1	0.2683
PPPP3	0.2533	PPPP3	0.3283
PPPP4	0.2454	PPPP4	0.2821
MMMM1	0.0498	MMMM1	0.0554
MMMM2	0.0759	MMMM2	0.0870
MMMM3	0.3019	MMMM3	0.3058
MMMM6	0.2921	MMMM6	0.3118
MMMM7	0.3156	MMMM7	0.3289
HDMNZLL	0.0584	HDMNZLL	0.0802
HDMNZLLH	0.1528	HDMNZLLH	0.1723
HDMNZLHH	0.2388	HDMNZLHH	0.2833
HDMNZHHH1	0.2630	HDMNZHHH1	0.3062
HDMNZHHH2	0.3155	HDMNZHHH2	0.3518

be aggregated through a set of combination of two results, which could be cumbersome. Data fusion has been applied in cross-language information retrieval [4, 3], recommendation systems [10], and many other areas.

8 Conclusion

In this paper, we have described an investigation on post-search data fusion techniques with the emphasis on combining ranked lists with diverse retrieval effectiveness. With various forms of related information available for a given topic, data fusion has been performed much more often to take the advantage of the available information. However, our studies have shown that the widely used data fusion method CombMNZ could not sustain its effectiveness when one or several ranked lists are low performance results. Through introducing predefined weights to boost singles from high quality ranked lists, and reduce the effect of noise from low performance ranked lists, our proposed methods WCombMNZ and WCombMWW can achieve significantly more reliable fusion results.

Our future work includes further experiments on more principle way to identify the weights associated with each ranked lists. Genetic algorithm and other quality indicators that have been explored in TREC Robust track experiments are the potential candidates. Another interesting direction is to examine the new method in data fusion with ranking information. What we have been examined were all on combining results by their scores. Lee’s study showed that fusion by ranking sometimes is as effective as fusion by scores. A third direction is to explore more about data fusion techniques in cross-language spoken document

Table 9. Experiments for WCombMNZ and WCombMWW on HARD runs.

Type	Fused runs	MAP
Training best	HDWMNZLL-2map	0.0697
Testing	HDWMNZLL-2map	0.0962
Training best	HDWMWLL-2map	0.0736
Testing	HDWMWLL-2map	0.0951
Training best	HDWMNZLLH-100map	0.2292
Testing	HDWMNZLLH-100map	0.2580
Training best	HDWMWLLH-20map	0.2444
Testing	HDWMWLLH-20map	0.2688
Training best	HDWMNZLHH-10map	0.2595
Testing	HDWMNZLHH-10map	0.3239
Training best	HDWMWLLHH-2map	0.2592
Testing	HDWMWLLHH-2map	0.3241
Training best	HDWMNZHHH1-5map	0.2658
Testing	HDWMNZHHH1-5map	0.3235
Training best	HDWMWHHH1-2map	0.2643
Testing	HDWMWHHH1-2map	0.3199
Training best	HDWMNZHHH2-map	0.3155
Testing	HDWMNZHHH2-5map	0.3503
Training best	HDWMWHHH2-map	0.3156
Testing	HDWMWHHH2-2map	0.3496

retrieval where evidence from different retrieval systems could be affected by quality of automatic speech recognition and that of machine translation. It would be useful to identify a quality prediction scheme for selecting the weights for data fusion in such situations.

References

- [1] James Allan. HARD Track Overview in TREC 2005. In *Proceeding of TREC 2005*, 2005.
- [2] N.J. Belkin, C. Cool, W.B. Croft, and J.P. Callan. The effect of multiple query representations on information retrieval system performance. In *Proceeding of SIGIR'93*, 1993.
- [3] Aitao Chen. Cross-language retrieval experiments at CLEF 2002. In *Proceedings of CLEF 2002*, pages 28–48, 2002.
- [4] Kareem Darwish and Douglas W. Oard. CLIR Experiments at Maryland for TREC 2002: Evidence Combination for Arabic-English Retrieval. In *Proceedings of TREC 2002*, pages 703–710, 2002.
- [5] E.A. Fox and J.A. Shaw. Combination of multiple searches. In *Proceedings TREC-2*, pages 243–252, 1994.
- [6] D.K. Harman. Relevance feedback revisited. In *Proceedings of ACM-SIGIR 92*, 1992.
- [7] Joon Ho Lee. Analyses of multiple evidence combination. In *Proceeding of SIGIR'97*, pages 267–276, 1997.

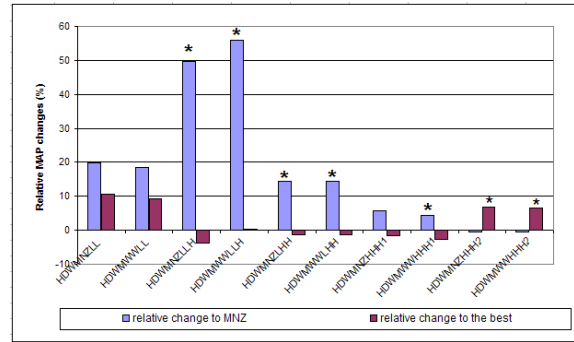


Figure 2. Relative MAP changes on HARD runs between WCombMNZ and WCombMWW and the two base-lines: corresponding CombMNZ runs and the best pre-fusion runs in the fusion process. All the runs related WCombMNZ contain “WMNZ” in their names, and those related to WCombMWW contain “WMWW”. The “*” on the top of a bar indicates that the change is statistically significant measured by paired t-test.

- [8] Douglas W. Oard, Dagobert Soergel, et. al. Building an information retrieval test collection for spontaneous conversational speech. In *Proceedings of SIGIR'94*, 2004.
- [9] M. Elena Renda and Umberto Straccia. Web metasearch: rank vs. score based rank aggregation methods. In *Proceedings of 2003 ACM Symposium on Applied Computing*, 2003.
- [10] Luis M. Rocha. Combination of evidence in recommendation systems characterized by distance functions. In *Proceedings of the 2002 World Congress on Computational Intelligence, FUZZ-IEEE'02*, pages 203–208. IEEE Press, 2002.
- [11] Ian Ruthven and Mounia Lalmas. Using Dempster-Shafer’s Theory of Evidence to Combine Aspects of Information Use. *Journal of Intelligent Information Systems*, 19:267–301, 2002.
- [12] G. Salton and C. Buckley. Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41(4):288–297, 1990.
- [13] Amanda Spink, Bernard J. Jansen, Chris Blakely, and Sherry Koshman. Overlap among major web search engines. In *ITNG '06: Proceedings of the Third International Conference on Information Technology: New Generations*, pages 370–374, Washington, DC, USA, 2006.
- [14] Theodora Tsirikla and Mounia Lalmas. Merging techniques for performing data fusion on the Web. In *Proceedings of CIKM'01*, 2001.
- [15] H. Turtle and W.B. Croft. Inference networks for document retrieval. In *Proceedings of SIGIR'90*, pages 1–24, 1990.
- [16] E.M. Voorhees. Overview of the TREC 2005 Robust Retrieval Track. In *Proceeding of TREC 2005*, 2005.