2007

Using OWA fuzzy operator to merge retrieval system results

H. Amiri
University of Tehran, Iran

A. AleAhmad
University of Tehran, Iran

Farhad Oroumchian
University of Wollongong in Dubai, farhado@uow.edu.au

C. Lucas
University of Tehran, Iran

M. Rahgozar
University of Tehran, Iran

Using OWA Fuzzy Operator to Merge Retrieval System Results

Hadi Amiri **Abolfazl AleAhmad** **Farhad Oroumchian** **Caro Lucas** **Masoud Rahgozar**

Department of ECE,University of Tehran
h.amiri@ece.ut.ac.ir,
a.aleahmad@ece.ut.ac.ir

University of Wollongong in Dubai
FarhadO@uow.edu.au

Department of ECE, University of Tehran
lucas@ipm.ir,
rahgozar@ut.ac.ir

Abstract

With rapid growth of information sources, it is essential to develop methods that retrieve most relevant information according to the user requirements. One way of improving the quality of retrieval is to use more than one retrieval engine and then merge the retrieved results and show a single ranked list to the user. There are studies that suggest combining the results of multiple search engines will improve ranking when these engine are treated as independent experts. In this study, we investigated performance of Persian retrieval by merging four different language modeling methods and two vector space models with Lnu.ltu and Lnc.btc weighting schemes. The experiments were conducted on a large Persian collection of news archives called Hamshari Collection. Different variations of the Ordered Weighted Average (OWA) fuzzy operators method, called a quantifier based OWA operator and a degree-of-importance based OWA operator method have been tested for merging the results. Our experimental results show that the OWA operators produce better precision and ranking in comparison with weaker retrieval methods. But in comparison with stronger retrieval models they only produce minimal improvements[1].

[1] This work was partially supported by Iranian Telecommunication Research Center (ITRC) contract No. 500/12204.

1 Introduction

The need for effective methods of automated information retrieval has increased because of the proliferation in amount of unstructured text data. Many approaches and methods have been developed to exhibit better retrieval engines (Witten et al., 1999; Singhal et al., 1996; Yates and Neto, 1999; Rijsbergen, 1997). In ad-hoc information retrieval many factors affect the effectiveness of methods, such as collection features, the method's algorithm and many other features. If we consider each one of retrieval systems as an expert to find related information, we can achieve better results.

In this research we considered retrieval as a fusion problem and we used a fuzzy OWA operator as a soft operator to merge the results of different ad-hoc retrieval engines. Moreover, we confirmed the previous results on Persian language that indicated the Lnu.ltu vector space method as one of the best retrieval engines on Persian text (Oroumchian and Garamalek, 2002; Garamalek, 2002). In this study, we considered four different language modeling methods proposed by Hiemstra (2001) and two vector space models for the fusion. The literature shows that these methods have good performance on TREC collections and outperform some other ad-hoc methods (Hiemstra, 2001; Ponte and Croft, 1998; Hiemstra, 2004; Larkey and Connell, 2002). The two vector space models use Lnu.ltu and Lnc.btc weighting schemes. The definitions of the used methods and the OWA operators are given in the subsequent sections.

The collection that is used in this study is a standard test collection of Persian text which is called Hamshahri (Darrudi et al., 2004). To merge the results of the retrieval engines and obtain a higher precision we have used two Ordered Weighted Average (OWA) fuzzy operators, a quantifier based OWA operator and a degree-of-importance based OWA operator. In quantifier based OWA operator, some linguistic quantifiers such as *All*, *Most*, *Few* and *At-Least-One* were used by OWA operator to merge the results. In degree-of-importance based OWA operator we used the position of the documents in the retrieved lists to construct the weighting vector of OWA operator. Our experimental results show that the best OWA operator, quantifier based OWA operator, improves the overall precision on average only by 0.003 over the best retrieval engine, the fourth model of Hiemstra (2001) , and 0.245 over the worst retrieval engine, the third model of Hiemstra (2001).

In section 2 we explain the two vector models and four Hiemstra's language modeling methods. In section 3 we define the OWA operators and the weighting schemas that are used in this research. The experimental results and comparisons are presented in section 4. Finally, the paper ends with the conclusions provided in section 6.

2 Definition of Retrieval Systems

This section provides a detailed description of the retrieval methods. Section 2.1 describes the vector space models with Lnu.ltu and Lnc.btc weighting schemes. Section 2.2 describes four different language modeling methods proposed by Hiemstra (2001).

2.1 Vector Space Models

Two vector space models used in the experiments are explained in this section. The first is the typical implementation which is also called Cosine Similarity. This model computes the cosine of the angle between the query and the document vector according to the following formula (Witten et al., 1999; Rijsbergen, 1997; Taghva et al., 2004).

$$\mathrm{L}nc = 1 + \log(tf_{d,t}) \,. \tag{1}$$

$$btc = \log(1 + \frac{N}{n}) \,. \tag{2}$$

In Lnu.ltu weighting schema, documents are weighted with Lnu and queries are weighted with ltu weighting. (Singhal et al., 1996)

$$\begin{aligned}\ln u = {} & [\frac{1+\log(tf)}{1+\log(avg\,(tf_d))}] * \\ & [\frac{1}{slope \;*\; NumberOfUniqueTerms} \\ & + (1 - slope) * pivot] \,.\end{aligned} \tag{3}$$

$$\begin{aligned}ltu = {} & [1+\log(tf)] * \log(\frac{N}{n}) * \\ & [\frac{1}{slope \;*\; NumberOfUniqueTerms} \\ & + (1 - slope) * pivot] \,.\end{aligned} \tag{4}$$

In the Equation 3 and 4, *Avg*(tf_d) is the average frequency of terms in the document and Slope is set to 0.25. This value is the best known tuned value for the *slope* parameter in vector models for Persian retrieval (Aleahmad, et al., 2007). Pivot is the Cosine Normalization which is formulated as:

$$Pivot = \sqrt{W_1^2 + W_2^2 + ... + W_n^2} \,. \tag{5}$$

Where W_i is the raw *tf*idf* weight for each term (Singhal et al., 1996).

2.2 Language Modeling Method- Hiemstra Method

In this section we describe four different language modeling methods proposed by Hiemstra (2001). Considering *P(D=d)* as the prior probability of relevance of the document *d*, the document that the user has in mind, to the query *q* with query terms $t_1,...,t_n$. The document *d* will be ranked by calculating the following probability:

$$\begin{aligned}& P(d, t1, t2, ..., tn) = P(D = d) \;\; * \\ & \prod_{i=1}^{n} ((1-\lambda i) \quad P(T = ti) + \lambda i \quad P(T = ti \mid D = d)) \,.\end{aligned} \tag{6}$$

Lambda (λ_i) is a smoothing parameter for each query term i (Hiemstra, 2001; Hiemstra, 2002). Hiemstra proposed four ways to specify the probabilities and parameters in Equation 6. He emphasizes in (Hiemstra, 2001) if there is no previous relevance information available for a query, i.e. none of the relevant documents have been identified yet, each query term that is not in the stop list, will be considered equally important. Hence, in this case the model has only one unknown parameter as λ_i will be equal for each position *i* in the query. The unknown parameter will simply be called λ in the new equation. So Equation 6 will be reformulated as below:

$$P(d, t1, t2, ..., tn) = P(D = d) \quad * \quad \prod_{i=i}^{n} ((1-\lambda) \quad P(T = ti) + \lambda \quad P(T = ti \mid D = d)) \quad (7)$$

Hiemstra proposed four different language modeling methods using Equation 7:

$$LM1 \quad (d) = \sum_{i=1}^{n} \log(1 + \frac{\lambda \quad tf(ti, d) \quad (\sum_t cf(t))}{(1-\lambda) \quad cf(ti) \quad (\sum_t tf(t, d))}) \quad (8)$$

$$LM2 \quad (d) = \sum_{i=1}^{n} \log(1 + \frac{\lambda \quad tf(ti, d) \quad (\sum_t df(t))}{(1-\lambda) \quad df(ti) \quad (\sum_t tf(t, d))}) \quad (9)$$

$$LM3 \quad (d) = \log(\sum_t tf(t, d)) + \sum_{i=1}^{n} \log(1 + \frac{\lambda \quad tf(ti, d) \quad (\sum_t cf(t))}{(1-\lambda) \quad cf(ti) \quad (\sum_t tf(t, d))}) \quad (10)$$

$$LM4 \quad (d) = \log(\sum_t tf(t, d)) + \sum_{i=1}^{n} \log(1 + \frac{\lambda \quad tf(ti, d) \quad (\sum_t df(t))}{(1-\lambda) \quad df(ti) \quad (\sum_t tf(t, d))}) \quad (11)$$

In the above equations, $tf(t_i, d)$ is the frequency of query term t in the document d and $cf(t)$ is collection frequency of query term t. $\sum_t tf(t, d)$ is the total number of terms in document d or length of document d, and $\sum_t cf(t)$ is total number of terms in the collection or collection length. $df(t)$ is document frequency of query term t and $\sum_t df(t)$ is defined by sum of document frequency for all terms in the collection which has a constant value (Hiemstra 2001). The differences between the four methods can be summarized as follows: Document frequencies are used instead of collection frequencies in LM 2 and LM 4. Document length correction is also added to LM 3 and LM 4. Hiemstra determined in a series of experiments that the *LM 4* was optimal for English text (Hiemstra 2001). We have implemented all of these four models on Persian text. For these experiments λ is considered 0.0485, the value that Taghva et al. (2004) determined as the optimal value of λ for Persian documents with no stemming and no stop word removal (Taghva et al., 2004).

3 OWA Fuzzy Operator

This section describes the Order Weighted Average (OWA) operator and two weighting schemas, quantifier-based weighting and degree-of-importance weighting.

3.1 OWA Definition

This section introduces the merge operator that we used for fusion. At first let us formalize the entities involved in the fusion problem. In the fusion problem the user query is indicated by q and n retrieval engines are indicated by $R_1, R_2, ..., R_n$. The ordered lists of documents produced by the retrieval engines for the query q are indicated by $L_{1,q}$, $L_{2,q}, ...,$ $L_{n,q}$. the purpose of the fusion is fusing the ordered lists and producing a unique ranked list for query q:

$$L_q = \text{Merge-Operator } (L_{1,q}, L_{2,q}, ..., L_{n,q}). \quad (12)$$

In Equation 12 L_q is constituted by a set of documents D, indicated by d_1, d_2, ..., d_k ordered by their degree of relevance to the query q.

In this research we used OWA operator as the merge operator. The OWA operator with n dimensions is a nonlinear aggregation operator OWA: $[0, 1]^n \rightarrow [0, 1]$ with a weighting vector $W=[w_1, w_2, ..., w_n]$ such that $\sum_{i=1}^{n} w_i = 1$ with w_i in [0, 1]. The OWA weight of each document d is defined as:

$$OWA(d) = OWA(x_1, x_2, ..., x_n). \quad (13)$$

where x_i indicated the score of document d in the ith list. Each score x_i is assigned by R_i to document d. If d is not present in the ith list then x_i=0 (Bordogna and Pasi, 2004; Callan, 2000). The OWA weight of each document is computed by Equation 14:

$$OWA(d) = OWA(x_1, x_2, .., x_n) = W^T \quad . \quad B = \sum_{i=1}^{n} w_i^T \quad * \quad bi \quad (14)$$

in which W^T is the transpose vector of W that defines the semantics of associated with the OWA operator and $B=[b_1, b_2, .., b_n]$ is the vector X=[x_1, $x_2, ..., x_n$] reordered so that $b_j = Min_j(x_1, x_2, ..., x_n)$, that is the jth smallest element of all the $x_1, x_2, ...,$ x_n. The values produced as a result of OWA operator aggregation lie between those produced by the *AND* (*Min Operator*) and those produced by the

OR (*Max Operator*) (Bordogna and Pasi, 2004; Yager, 1998; Yager and Kreinovich. 2002; Bordogna, et al., 2003).

As mentioned before the scores $\{x_i,\ i=1,...,n\}$ are assigned by the retrieval engines $R_1, R_2, ..., R_n$ to the document d, but there is a problem. These scores are not in the same scale. To address this problem we have used below function to bring the scores into a same scale (Callan, 2000):

$$\mu_i = \frac{x_i - Min(L_{i,q})}{Max(L_{i,q}) - Min(L_{i,q})} + c \ . \quad (15)$$

in which $Min(L_{i,q})$ and $Max(L_{i,q})$ are the minimum and maximum scores in the ith retrieved list for query q, $L_{i,q}$. This normalization normalizes the scores into the range [0, 1]. The value of c is set to 0.0001 to prevent $\mu_i = 0$ when $x_i = Min(L_{i,q})$. For all $\mu_i > 1$ we set $\mu_i = 1$.

Hence we can consider the new scores, $\{\mu_i,\ i=1,...,n\}$, as a membership degree of the documents in the retrieved lists for the queries. By this definition, the retrieval engines play the role of experts, the documents are the alternatives that are evaluated based on the user query and the decision function is a soft aggregation operator.

The OWA weight of a document d is computed using μ_i and the weighting schema discussed in the next section. Having OWA weights computed for all documents in the lists, we sort the documents according to their OWA weights and select top 20 of the merged results for each query.

3.2 Weighting Methods

In this section we discuss the weighting schemas that are used to merge the lists.

3.2.1 Quantifier Based Weighting

We used the linguistic quantifiers *All*, $Most_n$, Few_n, and A*t-Least-One* as the weighting schemas, to compute the overall performance judgment for each document with respect to a query. The quantifier *All* corresponds to the rigorous majority. This quantifier requires retrieving documents appearing in all retrieval engines' lists. This quantifier is suitable when the user is looking for precise answer since the set of results is formed by calculating the intersection of the retrieval engines lists (Bordogna and Pasi, 2004). The quantifier $Most_n$ is a fuzzy majority operator that assumes the retrieval by the most of the engines to be sufficient for inclusion in the fused list. The parameter n indicates the minimum number of retrieval lists sufficient for inclusion in the merge process. The quantifier Few_n is a weaker weighting schemas in which it is enough for a document to be retrieved by a few number of retrieval engines. The *At-Least-One* quantifier is the weakest weighting schemas in which it is enough for a document to appear in only one retrieval engine's list to be included in the fused list. This quantifier is suitable when the user wants to consider any potentially relevant document (Bordogna and Pasi, 2004). Hence the *All* quantifier has an *AND* semantic and the *At-Least-One* quantifier has an *OR* semantic (Bordogna and Pasi, 2004; Bordogna et al. 2003). The $Most_n$ and Few_n quantifiers have the semantics in between an *AND* and an OR operators.

3.2.2 Degree of Importance Based Weighting

As the second weighting schema we use the position of the documents in the retrieved lists to produce the weighting vector $W = [w_1, w2, ..., wn]$. The weight of each document d in the $L_{i,q}$ is defined by Equation 16:

$$w_i = \frac{N_i - POS_i - 1}{N_i} \ . \quad (16)$$

in which N_i is the number of elements in the ith list, $L_{i,q}$, and POS_i is the position of document d in $L_{i,q}$. With this definition we can think each w_i indicates the importance degree of a document with respect to a retrieval engine. This value decreases as the position of the document in the list goes from the top to the bottom of the list. We can use this degree to adjust the scores of the documents:

$$\mu_i = \mathrm{Max}(\mu_i, 1 - w_i) \ . \quad (17)$$

Using above equation to compute μ_i, the low score documents take a chance to contribute in the merged list especially if they are retrieved with the most of the retrieval engines. Hence the OWA weight of each document d is defined by:

$$OWA(d) = \sum_{i=1}^{n} w_i \ * \ \mu_i \ . \quad (18)$$

4 Experimental Results

This section describes the results of applying the quantifier based and the degree-of-importance based OWA operators on the defined six retrieval engines and merger of their results. In this study we used a standard test collection of Persian text which is called Hamshahri (Oroumchian and Darrudi, 2004). Hamshahri collection is the largest test collection of Persian text. This collection is prepared and distributed by University of Tehran. The third version of Hamshahri collection is 600MB in size and contains more than 160,000 distinct textual news articles in Persian. This collection has about 60 queries and relevance judgments for top 20 documents for each query. Older versions of this collection were used in other Persian information retrieval experiments (Garamalek, 2002; Bordogna and Pasi, 2004).

The Document Cut-Off criterion is selected to evaluate the precision of the six retrieval engines and the merge operators. Document cut-off diagram shows the precision after 5, 10, 15 and 20 documents have been retrieved. The values averaged over all queries. For the experiments we used the standard *TrecEval* tool which is provided by *NIST* (National Institution of Standards and Technology, 2006). Table 1 shows the precision of the six retrieval engines at different document cut-offs.

DOC	Lnc. btc	Lnu. ltu.25	LM1	LM2	LM3	LM4
5	0.342	0.556	0.488	0.559	0.329	0.570
10	0.375	0.593	0.498	0.563	0.358	0.588
15	0.388	0.599	0.513	0.575	0.361	0.606
20	0.389	0.604	0.505	0.571	0.355	0.604
AVG	**0.373**	**0.588**	**0.501**	**0.567**	**0.351**	**0.592**

Table 1 Performance of Retrieval Engines at different Document Cut-Offs

Figure 1 shows the document cut-off diagram. The X-axis represents the 4 document cut-offs and Y-axis shows the precision. The lowest precision value is 0.329 which belongs to LM3 model at 5 document cut-off.

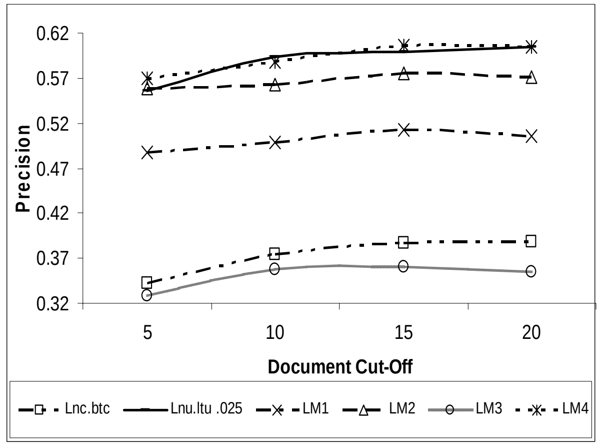

Figure 1 Document Cut-Off diagram of Retrieval Engines

The LM4 and the Lnu.ltu with slope 0.25 methods are better than the other systems. The LM3 is the worst method and the Lnc.btc is the second worst method.

As we mentioned before we used the OWA operator with two weighting methods to merge the retrieval engines' results. In quantifier based weighting schema each quantifier is described with a weighting vector $W=[w_1,w_2,\ldots,w_6]$ in witch w_i is the weight of the ith smallest element of all the scores $\{\mu_k,\ k=1,\ldots,n\ \}$ assigned to a document. Table 2 shows these vectors.

Method	Weighting Vector	Orness Degree
All	$W=[1,0,0,0,0,0]$	*0.00*
Most$_2$	$W=[0,.5,.5,0,0,0]$	*0.30*
Most$_3$	$W=[0,.33,.33,.33,0,0]$	*0.40*
Most$_4$	$W=[0,.25,.25,.25,.25,0]$	*0.50*
Few$_3$	$W=[0,0,.33,.33,.33,0]$	*0.59*
Few$_2$	$W=[0,0,0,.5,.5,0]$	*0.70*
At-Least-One	$W=[0,0,0,0,0,1]$	*1.00*

Table 2 Quantifier Based Weighting Vectors

In the above table the Orness Degree column is interpreted as an estimation of optimism of the weighting vector that is used by the OWA operator. The orness of the OWA operator is defined as below:

$$Orness(W) = (\frac{1}{n-1})\sum_{i=1}^{n}(i-1)*w_i\,. \qquad \textbf{(19)}$$

The orness degree is the maximum for the quantifier *At-Least-One* and is minimum for the *All* quantifier. The orness degree for the other quantifiers is a value in the range of (0,1) .

The weighting vectors are computed based on the definition of the quantifiers described in section 3.2.1. The weighting vector of quantifier *All* is *W*=[*1,0,0,0,0,0*]. This means the merge operator, OWA, is the AND operator that requires retrieving documents appearing in all retrieval engines' lists, hence the lowest score should be selected. The weighting vector of quantifier $Most_2$ is *W*=[0,*.5,.5,0,0,0*], in witch we average the 2th and 3th lowest *μ's* {μ_k, *k=1,…,n* } for each document. Hence in $Most_2$ only the 2th and 3th lowest scores are contributing to the merge operator. For the Few_i quantifiers it is sufficient that a document is retrieved by at least a minority of retrieval engines, hence we consider the high scores for each document. The weighting vector of quantifier *At-Least-One* is *W*=[*0,0,0,0,0,1*], in which for each document the highest score is considered as its weight.

Table 3 shows the results of OWA with quantifier based and degree of importance based schemas.

DOC	All	Most2	Most3	Most4	Few2	Few3	At-Least-One	DOI
5	0.53	0.60	0.57	0.55	0.56	0.55	0.47	0.58
10	0.42	0.60	0.60	0.60	0.59	0.59	0.49	0.60
15	0.34	0.55	0.60	0.60	0.59	0.59	0.48	0.58
20	0.28	0.50	0.59	0.60	0.59	0.59	0.48	0.57
AVG	**0.39**	**0.56**	**0.59**	**0.59**	**0.58**	**0.58**	**0.48**	**0.58**

Table 3 Performance of the Quantifier Based Weighting Schemas at different Document Cut-offs

As it is shown in Table 3, using OWA operator with the *All* quantifier produces a bad result, it is supposed to be bad and it is bad. The same is true for the *At-Least-One* quantifier. The better results are produced by the $Most_3$ and $Most_4$ *operators.* The precision of the degree of importance based schema, *DOI*, is one of the highest for cut-off 5 and 10 but it slightly decreases after retrieving more than 10 documents. Figure 2 shows the document cut-off diagram. The results of the methods that produced a precision value less than 0.5 are not depicted in Figure 2. Those are the *All* and *At-Least-One* quantifiers.

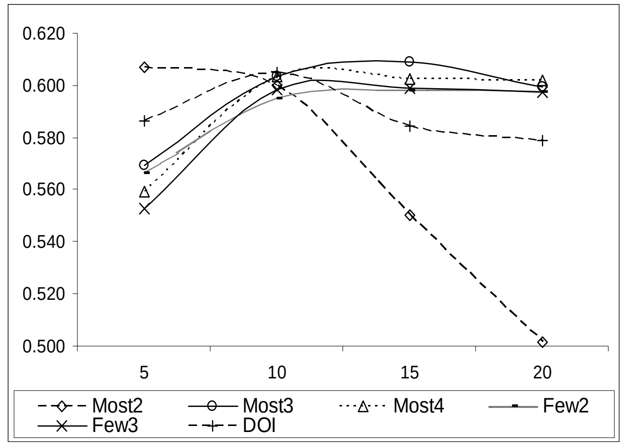

Figure 2 Document Cut-Off diagram of Quantifier Based Weighting Schema

In Figure2 the $Most_3$ and $Most_4$ quantifiers are better than the other systems at cut-off 10, 15, 20. The $Most_2$ has very good precision at cut-off 5 but its precision decreases afterwards. The *Few* quantifiers have low precision. We compared the results of $Most_3$ and $Most_4$ quantifiers with the LM4 and the Lnu.ltu 0.25 methods. We considered the LM4 and the Lnu.ltu with slope 0.25 methods because they demonstrate better performance than the other systems. Figure 3 shows this comparison.

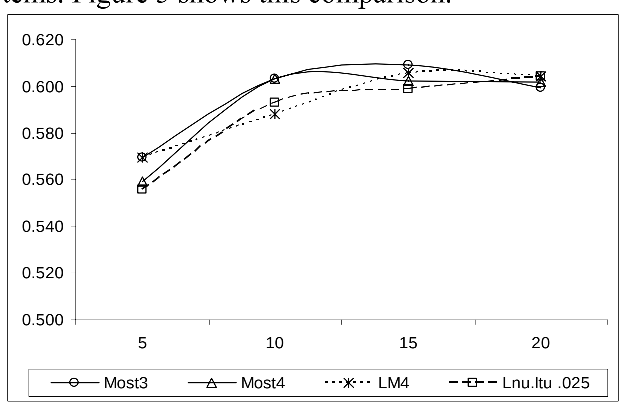

Figure 3 Document Cut-Off Diagram, Comparison of Method's Effectiveness

The $Most_3$ has the best precision at cut-off 5, 10 and 15, but its precision decreases for cut-off 20. Both LM4 and the Lnu.ltu have the highest precision at cut-off 20. Table 6 at Appendix A. summarizes the empirical result.

Two significance tests, namely T-Test and Wilcoxon Signed Rank Test have been performed on the results of $Most_3$, $Most_4$, LM4 and Lnu.ltu 0.25 methods in order to determine if the observed difference are the result of chance or not. For each of the four methods the query by query precision has been calculated and compared (Sanderson and Zobel, 2005) and then the significance tests have been conducted to see whether the null hypothesis that states the two groups do not differ, is rejected or not. Table 4 shows the result of the Wilcoxon Signed Rank Test.

Method	$Most_3$	$Most_4$	LM4
LM4	0.861	0.894	---
Lnu.ltu 0.25	0.081	0.077	0.444

Table 4 Wilcoxon Signed Rank SignificanceTest

Since the results are above 0.05 then the null hypothesis is rejected. Therefore the Mos_3 and $Most_4$ methods are better than the best single methods LM4 and Lnu.ltu with slope of 0.25. We then conducted a T-Test on the same models and the result of the T-test is depicted in Table 5.

Method	$Most_3$	$Most_4$	LM4
LM4	0.456	0.383	--
Lnu.ltu 0.25	0.028	0.027	0.147

Table 5 T-Test Statistical Significance Test Result

According to T-Test, if the *t* value is above the threshold chosen for statistical significance (usually the 0.05 level), the null hypothesis is rejected. Based on Table 5, both Mos_3 and $Most_4$ methods are significantly better than LM4 method which is a confirmation of The Wilconxon Signed Rank test. However, with the T-Test we can not confirm the significance of the Mos_3 and $Most_4$ methods over the Lnu.ltu with slope of 0.25 method.

5 Conclusion

In this study we looked at the Persian retrieval as a fusion problem. We used a fuzzy OWA operator as a soft operator to merge the results of different adhoc retrieval engines. For this purpose we considered four different language modeling methods proposed by Hiemstra (2001) and two vector space models, with Lnu.ltu and Lnc.btc weighting schemes that we know work well with Persian text. The collection that is used in this study is a standard test collection of Persian text which is called Hamshahri (Darrudi et al., 2004). To merge the results of the retrieval engines, we used Ordered Weighted Average (OWA) operators with quantifier based fuzzy weighting and degrees-of-importance based fuzzy weighting. In quantifier based OWA operator, some linguistic quantifiers, such as *All*, *Most*, *Few* and *At-Least-One* are used by OWA operator to merge the results. In degree-of-importance based OWA operator we used the position of the documents in the retrieved lists to construct the weighting vector of OWA operator. Our experimental results show that the best OWA operator, quantifier based OWA operator $Most_3$ *and* $Most_4$, only marginally improve over the best retrieval method on Persian text the LM4 methods. However seems they produce better ranking since they push the relevant documents to higher ranks. The significant tests we conducted seem to confirm that $Most_3$ *and* $Most_4$ are significantly better than all other methods but Lnu.ltu with slope of 0.25. However, the superiority over the Lnu.ltu with slope of 0.25 was not confirmed by T-Test.

References

Abolfazl Aleahmad, Parsia Hakimian, Farzad Mahdikhani and Farhad Oroumchian. 2007. *N-Gram and Local Context Analysis For Persian Text Retrieval*. International Symposium on Signal Processing and Its Applications, Sharjah U.A.E.

Amir Nayyeri and Farhad Oroumchian. 2006. *FuFaIR: a Fuzzy Farsi Information Retrieval System*. IEEE International Conference on Computer Systems and Applications.

Amit Singhal, Chris Buckley and Mandar Mitra. 1996. *Pivoted Document Length Normalization*. In Proc. of the 19th ACM SIGIR Conference.

C. J. van Rijsbergen. 1997. *Information Retrieval, second edition*. Butterworth Heinemann Publisher, Newton, MA, USA.

Djoerd Hiemstra. 2001. *Using Language Models for Information Retrieval*. PhD thesis, University of Twente.

Djoerd Hiemstra. 2002. *Term-Specific Smoothing for the Language Modeling Approach to Information Retrieval: The Importance of a Query Term*. In Proc. 25th ACM SIGIR conference.

Djoerd Hiemstra, Stephen Robertson and Hugo Zaragoza. 2004. *Parsimonious Language Models for Information Retrieval*. In Proc. 27th ACM SIGIR conference.

Ehsan Darrudi, Mohamad R. Hejazi and Farhad Oroumchian. 2004. *Assessment of a Modern Farsi Corpus*. The Second Workshop on Information Technology and its Disciplines, WITID2004.

Farhad Oroumchian and Firooz M. Garamalek. 2002. *An Evaluation of Retrieval Performance Using Farsi Text*.

Workshop on Knowledge Foraging for Dynamic Networking of Communities and Economies.

Farhad Oroumchian and Ehsan Darrudi. 2004. *Experiments with Persian Text Compression for Web*. WWW 2004.

Firooz M. Garamalek. 2002. *An Evaluation of Combinational Methods in Retrieving Persian Text*. Msc Thesis, Faculty of Engineering, University of Tehran.

Gloria Bordogna and Gabriella Pasi. 2004. *A model for a SOft Fusion of Information Accesses on the web. Journal of Fuzzy sets and systems,* 148(1):105-118.

Gloria Bordogna, Gabriella Pasi, Ronald R. Yager. 2003. *Soft approaches to distributed information retrieval*. *Journal of approximate reasoning*, 34(2):105-120.

Ian H. Witten, Alistair Moffat and Timothy C. Bell. 1999. *Managing Gigabytes: Compressing and Indexing Documents and Images*. Morgan Kaufmann Publishers, Los Altos, USA.

Jame Callan. 2000. *Distributed Information Retrieval*. In Advances in Information Retrieval: Recent Research from the Center for Intelligent Information Retrieval, W. Bruce Croft, ed. Kluwer Academic Publishers.

Jay M. Ponte and W. Bruce Croft. 1998. *A Language Modeling Approach to Information Retrieval*. In Proc. 21st ACM SIGIR Conference.

Kazem Taghva, Jeffrey Coombs, Ray Pareda and Tom Nartker. 2004. *Language Model-based Retrieval from Farsi Documents*. In Proc. ITCC 2004 Intl. Conference on Information Technology.

Leaj S. Larkey and Margaret E. Connell. 2002. *Arabic information retrieval at UMASS in trec-10*. In E.M Voorhees, D.K. Harman., The Tenth Text Retrieval Conference.

Mark Sanderson and Justin Zobel. 2005. *Information retrieval system evaluation: effort, sensitivity, and reliability*, In Proc. of the 28th ACM SIGIR conference.

National Institution of Standards and Technology: http://trec.nist.gov/trec_eval. November 2006.

Ricardo B. Yates and Berthier R. Neto. 1999. *Modern Information Retrieval*. Addison-Wisley Publisher, USA.

Ronald R. Yager. 1998. *On ordered weighted averaging aggregation operators in multi criteria decision making*. *IEEE Transactions on Systems, Man and Cybernetics*, 18(1):183- 190.

Ronald R. Yager and Vladik Kreinovich. 2002. *Main Ideas Behind OWA Lead to a Universal and Optimal Approximation Scheme*. Fuzzy Information Processing Society, Proceedings NAFIPS, Annual Meeting of the North American.

Appendix A. Glance View of the Empirical Results

Each row in Table 6 shows the number of relevant documents (of top 20) for each query and method. The average number of relevant documents for $Most_3$ and $Most_4$ is 12 and 12.05 respectively. These values are higher than those for LM4 (the best method), Lnu.ltu method and LM3 (the worst method). It shows that the Most operators retrieve more relevant documents in their retrieved lists. Additionally, Figure 3 shows that the $Most_3$ operator ranks relevant documents higher. So we can conclude that the $Most_3$ operator ranks more relevant documents in higher places in the final result.

QID	**LM4**	**Lnu.ltu**	**LM3**	**Most3***	**Most4**
1	6	6	7	**6**	6
2	12	9	9	**9**	10
3	12	15	3	**15**	12
4	15	17	6	**15**	16
5	14	14	9	**12**	12
6	7	9	1	**8**	8
7	18	18	17	**18**	18
8	10	13	6	**10**	11
9	2	2	1	**2**	2
10	13	12	13	**12**	13
11	18	19	10	**19**	19
12	7	2	3	**2**	2
13	19	18	16	**18**	18
14	17	17	16	**18**	18
15	17	11	7	**14**	14
16	13	14	1	**13**	14
17	16	18	10	**17**	16
18	11	10	4	**11**	11
19	12	15	6	**12**	12
20	7	12	7	**11**	11
21	8	7	7	**8**	7
22	18	13	1	**14**	16
23	19	16	15	**16**	14
24	5	10	3	**9**	9
25	11	11	7	**13**	14
26	16	14	14	**14**	14
27	17	13	5	**15**	16
28	10	13	6	**14**	14
29	18	17	17	**19**	19
30	12	12	4	**15**	13
31	18	12	4	**12**	13
32	10	11	7	**12**	12
33	19	18	4	**18**	18
34	17	13	6	**13**	15
35	15	15	8	**16**	15
36	14	10	4	**10**	13
37	11	8	1	**10**	11
38	15	14	5	**19**	18
39	19	16	4	**17**	17
40	13	11	11	**12**	12
41	9	8	7	**8**	7
42	10	5	4	**9**	8
43	11	12	5	**11**	11
44	8	10	3	**8**	7
45	16	16	8	**16**	16
46	12	11	9	**12**	13
47	10	8	6	**9**	10

48	9	10	11	**10**	10
49	7	7	5	**7**	7
50	14	15	9	**14**	15
51	6	8	8	**8**	8
52	9	11	1	**15**	16
53	4	6	4	**6**	6
54	7	6	6	**7**	7
55	4	6	3	**6**	5
56	7	6	2	**6**	7
57	9	12	9	**12**	12
58	9	11	9	**12**	10
59	14	12	12	**14**	13
AVG	**11.97**	**11.61**	**6.88**	**12.00**	**12.05**

Table 6 Empirical Results at a Glance