2007

N-gram and local context analysis for Persian text retrieval

A. AleAhmad
University of Tehran, Iran

P. Hakimian
University of Tehran, Iran

F. Mahdikhani
University of Tehran, Iran

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

N-GRAM AND LOCAL CONTEXT ANALYSIS FOR PERSIAN TEXT RETRIEVAL

Abolfazl Aleahmad [a], *Parsia Hakimian* [a], *Farzad Mahdikhani* [a], *Farhad Oroumchian* [a,b]
[a] Electrical and Computer Engineering Department, University of Tehran
[b] University of Wollongong in Dubai
FarhadOroumchian@uowdubai.ac.ae, {a.aleahmad, p.hakimian, f.mahdikhani}@ece.ut.ac.ir

ABSTRACT

The Persian language is one of the languages in Middle-East, so there are significant amount of Persian documents available on the Web. But there are relatively few studies on retrieval of Persian documents in the literature. In this experimental study, we assessed term and N-gram based vector space model and a query expansion method, namely, Local Context Analysis using different weighting schemes on a realistic corpus containing 160000+ news articles. Then we compared our results with previous works reported on Persian language. Our experimental results show that among the assessed methods, 4-gram based vector space model with Lnu.ltu weighting scheme has acceptable performance and Local Context Analysis has the best performance for Persian text retrieval so far.

1. INTRODUCTION

The Persian language (also know as Farsi) is one of the languages in the Middle East that is spoken in several countries like Iran, Tajikistan and Afghanistan. Persian uses Arabic like script for writing and consists of 32 characters that are written continuously from right to left. During its long history, the language has been influenced by other languages such as Arabic, Turkish, Kurdish, and even European languages such as English and French. Today's Persian (Farsi) contains many words from the above languages and in some cases these words still follow the grammar of their original languages in building plural, singular or different verb forms. Therefore, the morphological analyzers for this language need to deal with many forms of words that are not actually Farsi.

Although UNICODE has been adapted as national encoding standard however still there are many conventions for typing and storing (encoding) text among typists and digital distributors of Persian text. These complexities together, lead to difficulties in recognizing word boundaries and producing invalid words in the word extraction stage of an Indexer. On the other hand, there are indexing methods such as N-grams that are resilient against spelling errors or spelling variations [1]. Considering all those problems and lack of a robust morphological analyzer, one could speculate that N-gram based models could produce reasonable results for Persian text.

In this study, we investigated Local Context Analysis, unstemmed single term and N-gram based information retrieval using a standard retrieval model called vector space. We have implemented this model with different configurations and tested it on a standard Persian test collection.

Section 2 describes related works in this area, section 3 offers an overview of methods used in these experiments, section 4 details the experiments and their results and section 5 is the conclusion.

2. RELATED WORKS

Due to the special and different nature of the Persian language compared to other languages like English, the design of an information retrieval system in Farsi requires special considerations. Unfortunately, little efforts have been focused on this problem compared to other languages. Oroumchian et. al reported the result of a series of experiments conducted by applying the existing information retrieval techniques to Persian text [2]. They conclude that vector space model with *Lnu.ltu* weighting and unstemmed single words, performs better than other models and configurations.

In [3], authors have proposed the design and testing of a Fuzzy retrieval system for Farsi (FuFaIR) with support of Fuzzy quantifiers in its query language. Their comparison shows that the performance of FuFaIR is considerably better than that of vector space model. Also experiments in [4] suggest the usefulness of language modeling techniques for Farsi. Furthermore, the design and implementation of a Farsi stemmer is reported in [5]. Also in [6] the author has discussed and tested several weightings and methods on the Farsi language.

Our experiments differ from those reported in [2] and [6] in three major ways, first we used a larger test collection with more queries, second we have tested new methods like Local Context Analysis [7] and third we assessed Lnu.ltu scheme with two different slopes.

3. USED METHODS OVERVIEW

3.1. Vector Space Model

In this model, documents are represented as vectors of weighted terms and the similarity of the documents is measured by some function of their distance from each other in a multidimensional space. There are standard weighting schemes and naming conventions for calling

1-4244-0779-6/07/$20.00 ©2007 IEEE

each weighting configuration in this model [8]. Two weighting schemes that we selected for our experiments based on their reported performances on English and Persian are called *atc.atc* and *Lnu.ltu*. The *Lnu.ltu* scheme has a slope parameter which is obtained experimentally. We used two different values for slope parameter namely 0.25 and 0.75 with their associated document length normalization methods "*pivoted unique normalization*" and "*pivoted cosine normalization*" respectively. Those configurations have been reported to have the best performance in the vector space model [9].

3.2. Local Context Analysis (LCA)

Local Context Analysis is an automatic query expansion method that combines global analysis and local feedback and was introduced by Xu et. al in 1996 [7] and further improved in 2000 [10]. LCA is fully automated and there is no need to collect any information from user other than the original query, so it has many applications, for example Agichtein has used it to predict the extraction performance [11]. It was also compared with user relevance feedback in the TREC-8 Interactive Track Task and it was found to work better than methods that actually use relevance feedback from user [12].

LCA uses initially retrieved documents for query expansion. After the initial relevant documents are retrieved (retrieval method has to be a good information retrieval method), top *n* documents are broken into passages. No method of choosing the optimum number of passages is known but usually between 30 and 300 passages are used [8]. This approach is based on concepts, which are defined as single nouns, two adjacent nouns or three adjacent nouns instead of keywords or terms. These concepts are calculated in the local passages retrieved and then top *m* ranking concepts are chosen for query expansion. They are weighted with a linear function in the expanded query so that the first ranking concept has a weight of 1 and the *m*-th concept gets a weight around 0.1. Local Context Analysis is performed in three steps:

First, run the original query and retrieve the top ranked documents. Then break these documents into fixed length passages (300 words) and rank these passages as if they were documents.

Second, calculate similarity of each concept in the top ranked passages with the entire original query using similarity function *sim(q,c)*, calculated as:

$$sim(q,c) = \prod_{k_i \in q} (\delta + \frac{\log(f(c,k_i) \times idf_c)}{\log n})^{idf_i}$$

In which δ factor is a constant which prevents the similarity function from returning zero, usually it is near 0.1. The function $f(c,k_i)$ is calculated as:

$$f(c,k_i) = \sum_{j=1}^{n} pf_{i,j} \times pf_{c,j}$$

Where $pf_{i,j}$ and $pf_{c,j}$ are frequency of term k_i and concept c in j-th passage, respectively. In the *sim(q,c)* equation, inverse document frequency factors are calculated as below:

$$idf_i = \max(1, \frac{\log_{10}(N/np_i)}{5}), idf_c = \max(1, \frac{\log_{10}(N/np_c)}{5})$$

Where N is number of passages and np_i and np_c are number of passages that contain k_i and c, respectively.

Third, after these calculations, the top *m* ranked concepts are added to the original query and initial retrieval method is done with the expanded query. With exception that now the expanded query is weighted. The original query terms have a weight of 2 and the added concepts are ranked as $1 - (0.9 \times i) / m$, in which i is rank of the concept in concept ranking.

3.3. N-Gram method

N-grams are strings of length *N* generated from words in texts. In traditional vector space approaches, dimensions of the document space for a given collection of documents are words or sometimes phrases that occur in the collection. By contrast, in the N-gram approach, dimensions of the document space are N-grams, namely, strings of *N* consecutive characters extracted from words. Since number of possible strings of length *N* is a lot smaller than number of possible single words in a language, therefore N-gram approaches have smaller dimensionality [1]. So, N-gram method is a remarkably pure statistical approach, one that measures statistical properties of strings of text in a given collection without regard to the vocabulary, lexical or semantic properties of natural language(s) in which documents are written.

The N-gram length (*N*) and the method of extracting N-grams from documents vary from one author and application to another [13, 14].

4. OUR EXPERIMENTS AND THE RESULTS

For our experiments we used a standard test collection for Persian text which is called HAMSHAHRI [15]. HAMSHAHRI collection is the largest test collection for Persian text which is prepared and distributed by University of Tehran. In our experiments we used third version of HAMSHAHRI collection that is 600MB in size and contains about 160000+ distinct textual news articles in Farsi.

This collection has 60 queries with their relevance information. In order to find relevant documents, a pooling method has been employed with 5 different systems and top 20 documents retrieved by each query on each system have been judged. Older versions of this collection were used in other Farsi information retrieval experiments [3, 16].

4.1. Vector space model and LCA

In *Lnu.ltu* weighting scheme, documents are weighted with *Lnu* and user's query is weighted with *ltu* which are calculated as followed:

$$Lnu = \frac{\frac{1+\log(tf)}{1+\log(average(tf))}}{(slope\times N.U.T)+(1-slope)\times pivot}$$

$$ltu = \frac{\ln(tf)+1.0\times\ln N/n}{(slope\times N.U.T)+(1-slope)\times pivot}$$

In above formulas, *tf* is term frequency, *N* is number of documents in the whole collection, *n* is number of documents that the term occurs in them and *N.U.T* is number of unique terms within the specified document. *Slope* and *pivot* are constant variables that are used in this method which is called *pivoted normalization*. The *pivot* constant is average number of unique terms in the whole collection. For example in our term-based vector space model implementation, the sum of number of unique terms in each document was 30733694 , and the average was calculated by dividing it by number of documents 166774, hence *pivot* = 184.283. Singhal et. al reported in [9] different slopes and pivots in *Lnu.ltu* weighting scheme. Based on their study the following two configurations have the best performance:

- Slope=0.25 and using *pivoted unique normalization* (P.U.N.).
- Slope=0.75 and using *pivoted cosine normalization* (P.C.N.).

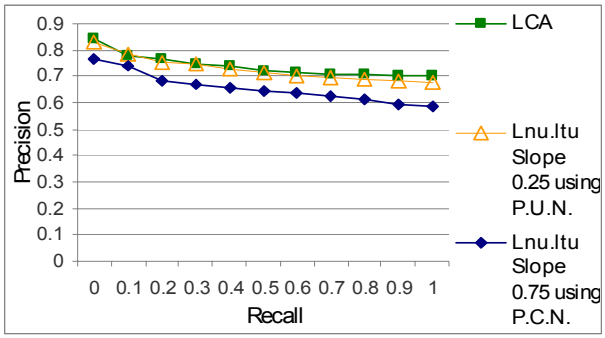

Figure 1. Interpolated Precision/Recall

Table 1. Interpolated Precision/Recall

Re-call	Precision		
	LCA	Lnu.ltu Slope=0.25	Lnu.ltu Slope=0.75
0	0.8457	0.8278	0.7632
0.1	0.7809	0.7867	0.7373
0.2	0.7691	0.7508	0.6857
0.3	0.7484	0.7444	0.6692
0.4	0.7405	0.7292	0.6558
0.5	0.7235	0.7166	0.6429
0.6	0.7161	0.6993	0.6353
0.7	0.7116	0.6941	0.6261
0.8	0.7086	0.6867	0.6096
0.9	0.7015	0.6819	0.5967

We utilized the above two configurations in our implementation of the vector space model. Table 1 and Figure 1 depict the interpolated precision and recall of the above two configurations when applied to the Persian text collection. The evaluation is done by using the standard *TrecEval* tool which is provided by *NIST* [17] and used in TREC evaluations. As it can be seen, *Lnu.ltu* model with slope=0.25 and *pivoted unique normalization* outperforms the same model with slope=0.75 and *pivoted cosine normalization*. This finding is consistent with what is reported for English text in [9].

In order to improve our results, we applied Local Context Analysis to the best system (*Lnu.ltu* with slope 0.25 and pivoted unique normalization) and expanded the original user's query using concepts. The expanded query is weighted using the described LCA weighting method. The number of retrieved passages was chosen as 20 and the size of each passage was 300 words based on what is reported in [7,18]. The number of chosen expanded query terms was 10 which means, we choose top 10 ranked concepts and added them to the original query. Figure1 and Table1 show that applying LCA as described above improves the precision even further by 2-3%.

4.2. N-gram

In the next part of our experiments, we assessed N-gram based vector space model for N = 3 and 4 on HAMSHAHRI collection using *Lnu.ltu* and *atc.atc* weighting schemes. It should be noted that we used N-grams that don't cross word boundaries. In case of *atc.atc* weighting scheme both user query and documents are weighted with *atc* that is calculated as below:

$$atc = 0.5+0.5\times\frac{tf}{\max tf}\times\ln\frac{N}{n}\times\frac{1}{\sqrt{\sum_i w_i^2}}$$

Where *N* is number of documents in whole collection, *n* is number of documents that contain *i*-th term and w_i is $tf \times idf$ for the *i*-th term in each document.

In case of *Lnu.ltu*, we used slope = 0.75 using *pivoted cosine normalization* as described above. Table 2 and Figure 2 depict interpolated precision and recall of the term based, 3-gram and 4-gram based vector space model with *Lnu.ltu* and *atc.atc* weighting schemes.

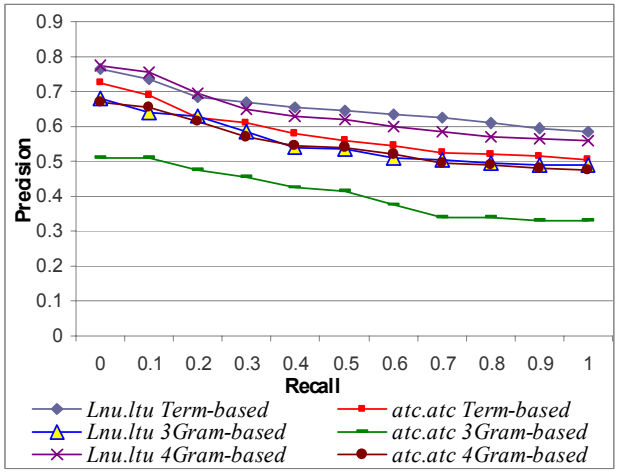

Figure 2. Interpolated Precision/Recall

As it is seen above, 4-gram based vector space with *Lnu.ltu* weighting scheme outperforms 3-grams and even term based configurations. This is in total contradiction with the performance of above model on English text where N-gram models are normally considered weak and only supplementary to the term based approaches. This is also better performance than we experienced before. The rational is that most Farsi words' roots are about 4 characters long. Since we are using a larger collection with more queries our current results are more valid than before.

Table 2. Interpolated Precision/Recall

Re-call	Precision					
	Term based		3-gram based		4-gram based	
	atc.ac	*Lnu.ltu*	*atc.ac*	*Lnu.ltu*	*atc.ac*	*Lnu.ltu*
0	0.7251	0.7632	0.5122	0.6783	0.6687	0.773
0.1	0.6876	0.7373	0.5122	0.642	0.6533	0.7571
0.2	0.6268	0.6857	0.476	0.6322	0.6156	0.6949
0.3	0.6094	0.6692	0.4543	0.5863	0.5676	0.6482
0.4	0.5802	0.6558	0.4257	0.5425	0.5436	0.6304
0.5	0.56	0.6429	0.4149	0.5363	0.5394	0.6211
0.6	0.5428	0.6353	0.3754	0.5116	0.5186	0.5996
0.7	0.5262	0.6261	0.3419	0.5039	0.4948	0.5846
0.8	0.5195	0.6096	0.3388	0.4966	0.4898	0.5721
0.9	0.513	0.5967	0.3316	0.4925	0.4804	0.5636

5. CONCLUSION AND FUTURE WORK

We have presented several experiments on Persian text using different configurations and we have proved that N-gram with N=4 are viable approaches that outperform their unstemmed term based counter parts. In comparison with the Fuzzy approach, the FuFaIR system [3], both the LCA system and the vector space model with *Lnu.ltu* weighting scheme and slope=0.25 and pivoted unique normalization have better performance than the FuFaIR system. Also, *Lnu.ltu* configurations have considerably better performance than atc.atc configurations and this is consistent with our previous findings on Persian text and also reported for English text in literature. In future, we want to assess 5-gram to see its performance and compare it with stemmed words.

Local Context Analysis only marginally improves the results over the *Lnu.ltu* method. This could be due to the fact that the *Lnu.ltu* weighting method is performing very well on the Farsi language and it is difficult to improve over its good result. But also it could be because we used LCA parameters that were used in TREC, and they might need adjustments for this collection. So, we need to run more experiments with different parameters to see if we can find a better configuration for the LCA method

REFERENCES

[1] W.B. Cavnar. "Using An N-Gram-Based Document Representation with A Vector Processing Retrieval Model". *In Proceedings of the Third Text REtrieval Conference (TREC-3)*, NIST Special Publication 500-225, pp. 269-277, 1995.

[2] F. Oroumchian, F. Mazhar Garamaleki. "An Evaluation of Retrieval performance Using Farsi Text". *First Eurasia Conference on Advances in Information and Communication Technology,* Tehran, Iran, October 2002.

[3] A. Nayyeri, F. Oroumchian. "FuFaIR: a Fuzzy Farsi Information Retrieval System". *IEEE International Conference on Computer Systems and Applications*, pp. 1126-1130, 2006.

[4] K. Taghva, J. Coombs, R. Pareda, T. Nartker. "Language Model-Based Retrieval for Farsi Documents". *International Conference on Information Technology: Coding and Computing (ITCC'04)*, 2004.

[5] K. Taghva, R. Beckley, M. Sadeh. "A Stemming Algorithm for the Farsi Language". *International Conference on Information Technology: Coding and Computing (ITCC 2005)*, 2005.

[6] F. Mazhar Garamaleki, *An Evaluation of Farsi Retrieval Method*, Msc Thesis, Department of Computer and Electrical Engineering, University of Tehran, 2003.

[7] J. Xu, W.B. Croft, "Query expansion using local and global document analysis". *In SIGIR '96 , Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. New York: ACM, pp. 4-11, 1996.

[8] Ed Greengrass, *Information retrieval: A survey.* DOD Technical Report: TR-R52-008-001, 2001.

[9] A. Singhal, C. Buckley, M.Mitra. "Pivoted Document Length Normalization", *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp21-29, 1996.

[10] J. Xu, W.B. Croft, "Improving the Effectiveness of information retrieval with Local Context Analysis". *ACM Transactions on information systems,* vol.18, No.1, pp. 79-112, January 2000.

[11] E. Agichtein, S. Cucerzan, "Predicting Extraction Performance by Using Context Language Models", *In the SIGIR 2005 Workshop on Methodologies and Evaluation of Lexical Cohesion Techniques in Real-World Applications*,2005.

[12] N.J. Belkin, C. Cool, J. Head, J. Jeng, D. Kelly, S. Lin, L. Lobash, S.Y. Park, P. Savage-Knepshield, C. Sikora . "Relevance Feedback versus Local Context Analysis as Term Suggestion Devices", *Rutgers' TREC-8 Interactive Track Experience*. 1999.

[13] S.H. Mustafa, "Character contiguity in N-gram-based word matching: the case for Arabic text searching", *Information Processing and Management,* Vol. 41, pp. 819–827, 2005.

[14] Y. Cebi, D. Dalkilic, "Turkish Word N-gram Analyzing Algorithms for a Large Scale Turkish Corpus – TurCo", *International Conference on Information Technology: Coding and Computing (ITCC'04)*, Vol. 2, p. 236, 2004.

[15] F. Oroumchian, E. Darrudi & M.R. Hejazi, "Assessment of a modern Farsi corpus", *Proceedings of The 2nd Workshop on Information Technology & its Disciplines (WITID)*, ITRC, Iran, 2004.

[16] F. Oroumchian, E. Darrudi, Experiments with Persian Text Compression for Web, WWW 2004, pp. 478-479, New York, New York, USA, May 2004.

[17] National Institution of Standards and Technology: http://trec.nist.gov/trec_eval/

[18] R.A. Baeza-Yates, B. Ribeiro-Neto, *Modern Information Retrieval*, Addison Wesley Pub Co Inc, ISBN: 020139829X, 1999.