Language Model-Based Retrieval for Farsi Documents

Kazem Taghva, Jeffrey Coombs, Ray Pareda, Tom Nartker
Information Science Research Institute
University of Nevada, Las Vegas

Abstract

This paper reports on an application of Language Modeling techniques to the retrieval of Farsi documents. We discovered that Language Modeling improves the precision of retrieval when compared to a standard vector space model.

1 Introduction

The Information Science Research Institute (ISRI) at the University of Nevada, Las Vegas has recently developed several new technologies to aid in the retrieval of Farsi documents.[12] In particular, we have created a stemming algorithm and a stopword list for Farsi.[10, 11]

Although stemming and stopword removal improved the average precision of retrieval for our Farsi collection[11], we wondered whether language model-based retrieval, which has been applied with success to Arabic documents[7], might not also improve Farsi retrieval. We have found that doing so can provide substantial improvement compared to a standard vector space model.

In this paper we present the basic concepts of language model-based and vector space information retrieval, a brief review of the concepts of stemming and stopword removal, a description of our experimental design, our results, and finally prospects for future study.

2 Language Model-Based Retrieval

Recently statistical language modeling has been applied extensively to the document retrieval problem.[9, 1, 8, 5, 14] Statistical language modeling was first used by Andrei Markov to model sequences of letters in the Russian language. Language modeling has since been applied with success to automatic speech recognition and natural language processing, as well as many other uses.[5]

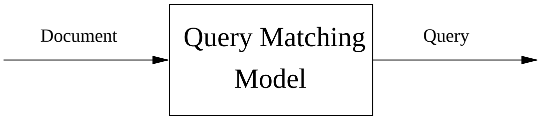

Figure 1. Information Theoretic Retrieval

One way to understand statistical language modeling is to conceive of the document retrieval problem in *information theoretic* terms.[9, 1, 14] According to this view, document retrieval is represented with the *source channel model* used in information theory.[2] Thus, the retrieval process is considered to be a decoding of a document transmitted over a noisy information stream, as illustrated in figure 1. In the case of information retrieval, the noisy information channel is the mind of the user, which tries to imagine what document it desires.[8] The query the user formulates becomes the only evidence of what the original message (document) is. The retrieval system, on this approach, is a function mapping query terms to the document such that the fuction attempts to decode the query terms back into the original document. The optimal function will retrieve the document with the highest probability given the query terms, i.e.,

$$f(q_1, \cdots, q_n) = \arg\max_d P(D|Q_1, \cdots, Q_n)$$

Here $Q_1, \cdots, Q_n$ are query terms, and D is a given document.

Using Bayes' rule and dropping the denominator which remains constant over documents:

$$f(q_1, \cdots, q_n) = \arg\max_d P(Q_1, \cdots, Q_n|D)P(D)$$

The probability distribution $P(D)$ is the *language model* of documents in a given collection. It describes how probable a document is in our language. Since the document is the 'source' in our model, the distribution

$P(Q_1, \cdots, Q_n|D)$ models the ’channel’, i.e., the mind of the user formulating the query.

These probability distributions are typically estimated with the *maximum likelihood* (ML) estimate. For example, we will estimate $P(D)$ as

$$P(D) = \frac{\sum_t tf(t,d)}{\sum_{t,k} tf(t,k)} \tag{1}$$

where $tf(t,d)$ is the *term frequency* of term t in document d, that is, then number of times t occurs in d. In the demoninator, we sum over all term frequencies in all documents.

However, the problem with maximum likelihood estimates is that potentially important items which are not represented in training data are assigned a probability of zero. For example, consider the problem of speech recognition. In speech recognition, the English representation of a series of sound waves is determined by a probability as above. However, training of such systems usually consists of only a small subset of the terms the software will have to recognize. As a result, two probability distributions are required, the probability that a sound corresponds to a ’seen’ word and the probability that is corresponds to a term it has not been trained on, an ’unseen’ term.[2, 14] In order for such a system to function on ’unseen’ data some small portion of the probability space must be assigned to such data.

This process of assigning some weight to unseen data is called *smoothing*. In information retrieval, smoothing will assign some weight to a document in the collection even if a given query term does not appear in the document. Letting λ_i be the weight given to seen (query) terms and $1 - \lambda_i$ the weight for unseen terms, relative to document D_i we rank documents in terms of the probability:

$$P(D)\prod_{i=1}^{n}((1-\lambda_i)P(Q_i) + \lambda_i P(Q_i|D)) \tag{2}$$

Although there are several methods for defining each of these probabilities, Djeord Hiemstra[5] determined in a series of experiments that the following definitions were optimal:

$$\begin{aligned} P(D) &= \tfrac{\sum_t tf(t,d)}{\sum_{t,k} tf(t,k)} \\ P(Q_i|D) &= \tfrac{tf(t_i,d)}{\sum_t tf(t,d)} \\ P(Q_i) &= \tfrac{df(t_i)}{\sum_t df(t)} \end{aligned}$$

Here, $tf(t_i, d)$ is the number of occurrences of term t_i in document d, that is, the *term frequency* of t_i, and $df(t_i)$ is the number of documents in the collection in which term t_i appears. This count is usually called the *document frequency* of term t_i.

Dividing equation 2 by the expression $\prod_{i=1}^{n}((1-\lambda_i)P(Q_i))$ will not affect the ranking since λ_i and $P(Q_i)$ have the same value for each document d. For the same reason, we may ignore the denominator of $P(D)$ and define it as: $P(D) = \sum_t tf(t,d)$.[5] Finally, replacing the product of probabilities with the sum of logs of probabilities gives us a formula which is easily implemented:

$$\begin{aligned} HLM4(d) = \log(\textstyle\sum_t tf(t,d)) + \\ \textstyle\sum_{i=1}^{n} \log(1 + \frac{\lambda_i tf(t_i,d)(\sum_t df(t))}{(1-\lambda_i)df(t_i)(\sum_t tf(t,d))} \end{aligned} \tag{3}$$

We call equation 3 *HLM4* since it is Hiemstra’s fourth version of his Language Model equation 2.

3 Vector Space Model

The Language Model was compared in our experiments to a typical implementation of the *vector space model* using the cosine similarity measure.[13] In the vector space model, each document and query is represented as a vector of term weights.

The cosine similarity measure derives its name from the well-known formula for the dot product of two vectors

$$Q \cdot D = |Q||D| \cos\theta$$

where Q is now a vector consisting of weights assigned for each term in the query, and D is similarly a vector of weights for each term in the document.

Hence the cosine of the angle between a query vector and a document vector can be defined as:

$$\begin{aligned} \cos(Q,D) &= \tfrac{Q \cdot D}{|Q||D|} \\ &= \tfrac{\sum_{t=1}^{n} w_{q,t} \cdot w_{d,t}}{W_q W_d} \end{aligned}$$

where W_d is the Euclidean length or weight of a document defined as:

$$W_d = \sqrt{\sum_{t=1}^{n} w_{d,t}^2}$$

The Euclidean length or weight of the query will be:

$$W_q = \sqrt{\sum_{t=1}^{n} w_{q,t}^2}$$

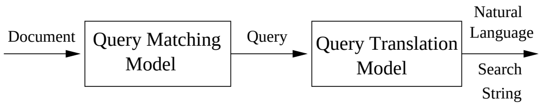

Figure 2. Two Level Model of Information Theoretic Retrieval

The expression $w_{q,t}$ represents the weight of the query term t and is typically defined as:

$$w_{q,t} = 1 \cdot \log\left(1 + \frac{N}{df_t}\right)$$

And $w_{d,t}$ is the weight of a document term:

$$w_{d,t} = 1 + \log(tf_{t,d})$$

Because the weights of terms are defined using term frequency tf and the inverse of document frequency df, the vector space model is often called the $tfidf$ model for retrieval.[13]

4 Stemming and Stopword Removal

Farsi, also known as Persian, is the official language of Iran and, along with Pashsto, one of the official languages of Afghanistan. It is also spoken in Tajikistan and parts of Uzbekistan [6]. There are more than two million Iranians living in the United States, and Farsi is taught at universities and institutions throughout North America and Europe [3]. In recent years, many web sites have appeared that provide information in Farsi. In particular, most of Iranian newspapers and magazines have official websites with daily articles in Farsi. As more Farsi materials become available on the web, it is evident that search tools need to be developed to better access Farsi information sources.

One aspect of language modeling not discussed in our presentation in section 2 is the distinction between (i) the query conceived of as a corrupted encoding of a document and (ii) the natural language sequence of search terms formulated by a user to represent the query. Hence, there is a second level of decoding required for document retrieval: a mapping from the natural language terms formulated by the user and the 'abstract' query, which in turn is a mapping to the document. The two level model is illustrated in figure 2.[5, 1]

As in traditional approaches to information retrieval, we adopt a one-many approach to the query translation task using the two methods of stemming and stopword removal. Stemming is the process whereby morphological variants of a term stem are mapped to that stem. Examples of morphological variants of the English term *stop* include *stops*, *stopping*, and *stopped*.[4] Stopwords are terms that carry little information generally due to the high frequency of their occurrence. Examples in English are *the*, *is*, and *this*. A small number of terms can account for 25% of the memory in a retrieval engine without contributing to its effectiveness.[13]

Since stemming and stopword removal are language-specific tasks, and no Farsi stemmers or stopword lists were available, ISRI had to develop both. Details of their construction have been reported elsewhere[10, 11] and are available on the Internet.[1]

	Cosine	HLM4	% change
recall	precision	precision	
0	0.435	0.478	9.885
10	0.374	0.422	12.834
20	0.313	0.363	15.974
30	0.253	0.291	15.020
40	0.201	0.213	05.970
50	0.179	0.193	07.821
60	0.158	0.168	06.329
70	0.128	0.148	15.625
80	0.110	0.110	00.000
90	0.088	0.097	10.227
100	0.084	0.091	08.333
Average	0.211	0.234	10.900

Figure 3. HLM4 vs. Cosine

5 Testing

The text collection used for testing consists of 1647 documents collected from Farsi websites. Sixty queries were formulated by Farsi language experts at ISRI and relevant documents assigned for each query.[11]

The collection was randomly split into two parts: a validation set of 824 documents and a test set of 823. The validation set is required for the language-model approach in order to estimate an optimal value for the parameter λ_i in equation 3.[5] Rather than determining λ_i separately for each document, a "global" λ was estimated by running the queries on the validation set several times while varying λ. The λ which produced the highest average recall (defined below in section 6) was selected as the optimal value. The optimal value for the

[1] `www.isri.unlv.edu/publications/isri-tech.php`

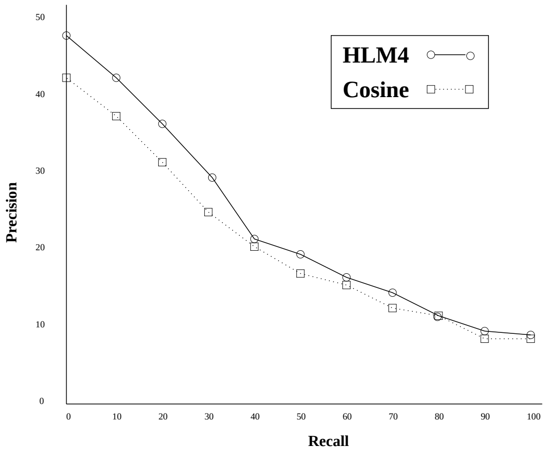

Figure 4. Comparison of Cosine Vector Model and HLM4 Langauge Model

collection using stemming and with stopwords removed was found to be $\lambda = 0.035$.

6 Results

Eleven point recall and precision statisitics are presented in figure 3. Recall is the number of relevant documents retrieved divided by the total number of relevant documents. Precision is the number of relevant documents retrieved at a given point divided by the total number of retrieved documents. For our results, an *interpolated* precision was calculated. For each query, the interpolated precision is the precision at a given recall level and all higher levels.[13] Thus, on the eleven point table in figure 3, among the first ten percent of documents retrieved (the row where recall is 0) 43.5% of the documents were relevant for the cosine procedure while 47.8% were relevant for the HLM4 language-model. Overall the language model approach improves precision by an average of nearly 11%. The eleven point table is presented graphically in figure 4.

In earlier experiments we had shown for the vector space model that our stemming and stopword removal algorithms improved retrieval.[11] The same holds for language-modeling. The same Farsi document set was indexed without stemming or stopword removal. Again, a series of tests were run on the validation document set to determine the optimal λ for formula 3. We determined that $\lambda = 0.0485$.

	HLM4-SS	HLM4-NSS	% change
recall	precision	precision	
0	0.478	0.459	2.027
10	0.422	0.403	2.303
20	0.363	0.326	5.370
30	0.291	0.244	8.785
40	0.213	0.206	1.671
50	0.193	0.183	2.660
60	0.168	0.147	6.667
70	0.148	0.131	6.093
80	0.110	0.110	0.000
90	0.097	0.104	-3.483
100	0.091	0.103	-6.186
Average	0.234	0.220	3.08

Figure 5. HLM4-SS vs. HLM4-SSN

The eleven point table in figure 5 compares retrieval using HLM4 with stopword removal and stemming, called HLM4-SS, to HLM4 including stopwords and with no stemming, called HLM4-NSS. HLM4-SS improves retrieval on average by about 3%. HLM4-NSS interestingly seems to do better at high recall levels than HLM4-SS. However, since users are typically only interested in the top 10% of results, i.e. low recall documents, it is probably better to improve results at the low recall end of our table than at the high end. Figure 6 graphically presents the eleven point results.

7 Conclusion

The table in figure 7 summarizes the results of our tests with the overall average of the eleven point precision for all four tests. This figure indicates that the worst approach for retrieving Farsi documents is to use the vector space model without stemming or stopword removal. The best results were obtained by using stemming and stopword removal with the language-model approach as defined by formula 3.

8 Further Study

Future research should include an evaluation of our assumption that equation 3 is the best interpretation of equation 2. Hiemstra offers four possible formulations of equation 2 and supported equation 3 on the basis of his own experiments.[5] Those experiments were on English texts, and it would be interesting to know if a Farsi document collection would behave similarly.

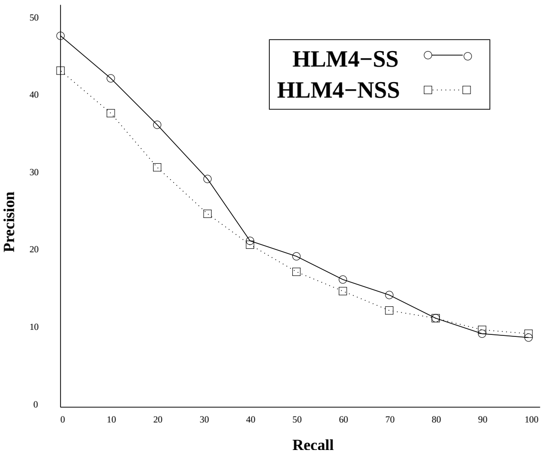

Figure 6. Effects of Stemming and Stopword Removal for HLM4

Cosine-NSS	Cosine-SS	HLM4-NSS	HLM4-SS
0 .180	0.211	0.220	0.234

Figure 7. Eleven Point Average Precision Comparision

References

[1] A. Berger and J. Lafferty. Information retrieval as statistical translation. In *Proceedings of the 22nd ACM Conference on Research and Development in Information Retrieval (SIGIR'99)*, pages 222–229, 1999.

[2] Stephen F. Chen. *Building Probabilistic Models for Natural Language*. PhD thesis, Harvard University, 1996.

[3] Center for Advanced Research on Language Acquisition (CARLA). Less commonly taught languages (LCTL) project. `http://carla.acad.umn.edu/LCTL/`.

[4] William B. Frakes. Stemming algorithms. In William B. Frakes and Ricardo Baeza-Yates, editors, *Information Retrieval: Data Structures and Algorithms*, pages 131–160. 1992.

[5] Djoerd Hiemstra. *Using Language Models for Information Retrieval*. PhD thesis, Universiteit Twente, 2000.

[6] Tore Kjeilen, Sidahmed Abubakr, and D. Josiya Negahban. Encyclopedia of the orient. `http://i-cias.com/e.o/`.

[7] Leah Larkey and Margaret Connell. Arabic information retrieval at umass in trec-10. In E.M Voorhees and D.K. Harman., editors, *The Tenth Text Retrieval Conference, TREC 2001 NIST Special Publication 500-250*, pages 562–570, 2002.

[8] D.R.H. Miller, T. Leek, and R.M.Swartz. A hidden Markov model information retrieval system. In *Proceedings of the 22nd ACM Conference on Research and Development in Information Retrieval (SIGIR'99)*, pages 214–221, 1999.

[9] Jay Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In *SIGIR '98*, 1998.

[10] Kazem Taghva, Russell Beckley, and Mohammad Sadeh. A list of farsi stopwords. Technical Report 2003-01, Information Science Research Institute, University of Nevada, Las Vegas, July 2003.

[11] Kazem Taghva, Russell Beckley, and Mohammad Sadeh. A stemming algorithm for the farsi language. Technical Report 2003-02, Information Science Research Institute, University of Nevada, Las Vegas, August 2003.

[12] Kazem Taghva, Ron Young, Jeff Coombs, Ray Pereda, Russell Beckley, and Mohammad Sadeh. Farsi searching and display technologies. In *Proc. of the 2003 Symp. on Document Image Understanding Technology*, pages 41–46, Greenbelt, MD, April 2003.

[13] I. Witten, A. Moffat, and T. Bell. *Managing Gigabytes*. Morgan Kaufmann, 2nd edition, 1999.

[14] Chengxiang Zhai and John Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of the 24nd ACM Conference on Research and Development in Information Retrieval (SIGIR'01*, pages 334–342S, 2001.