International Journal of Information Science and Management
Vol. 14, No. 2, 2016, 107-117

PHMM: Stemming on Persian Texts using Statistical Stemmer Based on Hidden Markov Model

Fatemeh Momenipour
Department of Computer Engineering,
Islamic Azad University, Qazvin Branch,
Qazvin, Iran
sonaymomeni@gmail.com

Mohammad Reza Keyvanpour
Department of Computer Engineering,
Alzahra University, Tehran, Iran
keyvanpour@alzahra.ac.ir

Abstract

Stemming is the process of finding the main morpheme of a word and it is used in natural language processing, text mining and information retrieval systems. A stemmer extracts the stem of the words. Persian stemmers are classified into three main classes: structural stemmers, dictionary based stemmers, and statistical stemmers. The precision of structural stemmers is low and the expenses of dictionary based stemmers is high; therefore, the main goal of this research was to design and implement a statistical stemmer based on Hidden Markov Model with high precision in order to reduce the size of indexed file and increase the speed of information retrieval systems. In the present study, the proposed stemmer finds the prefixes and suffixes of a word and removes them, so that the rest of the word is considered to be the stem. But there are some exceptions in Persian words which would be considered as a stem mistakenly. So, at first a dictionary of Persian stemmers was collected and after that the proposed stemmer searched a word in the dictionary, if the word was not there, the stemmer found the stem of it by HMM based stemmer. This stemmer was tested in Bijankhan corpus and Hamshahri test collection. The results showed increment in mean average precision and recall. The speed of the Information retrieval system was increased and the size of indexed files were decreased by the algorithm.

Keywords: Stem, Stemmers, Hidden Markov Model, Persian Words.

Introduction

In Persian languages, the main part of a word is its stem. The prefixes and suffixes are added to stems to change grammatical rule of the word (Mahdavi, 2015). Stemmer is a software which reduces all forms of words to the same morphological root, which is called "stem" (Rahimtoroghi, Faili & Shakery, 2010; Estahbanati, Javidan, & Nikkhah, 2011; Estahbanati, & Javidan, 2011). Farsi is an Indo-European language which is spoken and written in Iran, some part of Tajikistan and Afghanistan. It is written from right to left, so, suffixes are added to the left side of the word and prefixes are added to the right. These Affixes are added to the nouns to change their meaning, plurality and possession (Mokhtaripour, & Jahanpour, 2006; Sharifloo, & Shamsfard, 2008). Stemmers are used in

information retrieval system to improve the precision and recall (*Mehrad, & Koleini, 2007*). It also decreases the size of indexed file, and subsequently increases the speed of the system (Rahimtoroghi *et al.,* 2010; Taghva, Beckley, & Sadeh, 2005). There are three main approaches for stemming words in Persian. The first approach is the structural stemmer, which uses the structural rules of words for stemming. This kind of stemmer needs an expert person to design it and it is dependent on the language under study (Jadidinejad, Mahmoudi, & Dehdari, 2010). The second approach is the dictionary based stemmer in which all forms of the words and their stems are saved in a dictionary. The precision of this kind of stemmer is high but it needs to be updated repeatedly. The last approach is the statistical stemmer which finds the stem of the words by statistical rules and machine learning methods. This approach doesn't need to know the morphological rules of the words and also it is not dependent to language (Mohammad Nasiri, Sheikh Esmaeili, & Abolhassani, 2006; Momenipour, Moghadam, & Keyvanpour, 2013).

In this paper a statistical Persian stemmer based on Hidden Markov Model is proposed. This model uses strong mathematical rules to recognize pattern of a model. The rest of the paper is as follows: section two presents a brief review of previous work in the field. In section three Hidden Markov Model is explained in detail. Section four is an explanation of our newly proposed algorithm in details. Section five presents the experiments and results, and finally in section six, conclusion and future work are presented.

Previous Work

There are three main approaches for Persian stemming: the structural approach that uses the morphological structure to find the stem of every word; Lookup table, which saves each word and all related forms of them in a database, so that in order to find the stems, the words are to be searched in the database (Krovetz, 1993); and the last approach is statistical methods, in which statistical and machine learning techniques are used (Melucci and Orio, 2003). Rahimtoroghi, Faili, and Shakery (2010) presented a structural stemmer which used some heuristic rules based on the structure of Persian language and all its exceptions. They made 33 rules to remove the affixes of words. They tried to find the stem of nouns, adjectives and adverbs and didn't work on verbs, because they believed that users usually don't use verbs in their query. In the study by Mehrad and Berenjian (2011) made use of 10 suffixes and almost 2000 exceptions to make the RICesT stemmer. Lexical, structural and syntactical process were applied to find the stem of words. Their proposed stemmer had some advantages: it could find different forms of a noun, and also automatically classify texts in large files. Users can personalize the system according to their need and apply the results. They also can make a statistical report of their work. This stemmer was installed in the Regional information center for science and technology for the first time (Mehrad, & Naseri, 2010). In Mohammad Nasiri, *et al.,* (2006) a statistical Persian stemmer was presented based on Bacchin algorithm. In the learning phase of this stemmer, system finds lists of the affixes of words and weights them; in the testing Phase, system finds all the substrings of a word by use of lists of affixes which were found in learning phase. They used various states of words and found best stems of them. The designers of this stemmer believed their stemmer worked well. But the results showed that although this model was more complicated than structural

stemmers, it didn't reach a higher improvement (Rahimtoroghi *et al.*, 2010; Estahbanati *et al.*, 2011).

Hidden Markov Model

The newly developed Persian stemmer proposed in this paper uses statistical method based on Hidden Markov Model. HMM is a tool which represents distributions over sequences of observations (Rabiner, 1983; Song, Boots, Sajid, Gordon, & Smola, 2010). In HMM, transition between states are defined by probability functions and a symbol is emitted at each state by a given probability (Ghahramani, 2002).

According to Ghayoomi (2012) and Melucci and Orio (2003) an HMM is defined with the use of some parameters as follows:

1. N, is the number of states of the model.
2. M, is the number of symbols which is produced at each state.
3. π, is the initial state probability:

$$\Pi = \{\pi_i\},\ 1 \le i \le N \tag{1}$$

4. A, is the state transition probability distribution

$$a_{ij} = p[q_{t+1} = S_j | q_t = s_i] \qquad 1 \le i,j \le N \tag{2}$$

Where q_t is the current state and qt_{+1} is the next state probability so a_{ij} is the probability of transition from state i to state j. a_{ij} has some properties:

$$a_{ij} \ge 0\ ,1 \le i\ ,j \le N \tag{3}$$

$$\sum_{j=1}^{N} a_{ij} = 1\ ,1 \le i \le N \tag{4}$$

5. B, is the observation symbol probability distribution:

$$B_j(k) = p[v_k \text{ at } t | q_t = s_j] \quad 1 \le j \le N\ ,\ 1 \le k \le M \tag{5}$$

In formula 5, v_k is the k^{th} observable symbol. So $B_j(k)$ is the probability of producing v_k where model is in state $q_{i.}$. $B_j(k)$ has some properties:

$$b_j(k) \ge 0,\ 1 \le j \le N,\ 1 \le k \le M \tag{6}$$

So we can use the following notation for hmm model:

$$\lambda = (\pi, A, B) \tag{7}$$

Proposed Method

Hidden Markov Model is used to design a statistical Persian stemmer in this paper. HMM is used in modeling process which is unknown, but can be observed by a sequence of symbols (Melucci, & Orio, 2003). There are three kinds of words in Persian: verbs, nouns and pronouns. Pronouns do not have stem. In our proposed algorithm, Persian Hidden Markov Model (PHMM), there are three states for a word: prefix, stem and suffix. The sequences of

letters which create a word, are the observations of the model. There are some rules in PHMM:

- In Farsi language, a word never begins with a suffix. So, the third element of initial state probability vector is zero.
- Transition from suffix state to prefix and stem state, and also transition from stem state to prefix state are zero.

The topology of PHMM is shown in Figure 1.

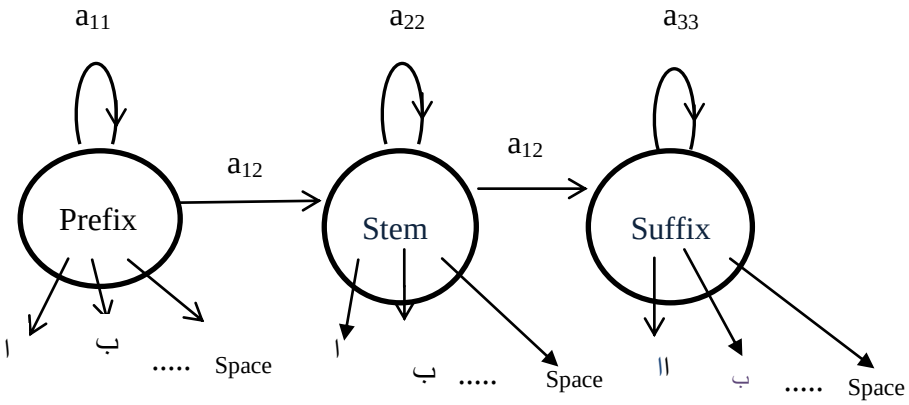

Figure1: The topology of PHMM

The first step of PHMM is to recognize its parameters. To estimate these parameters some words from Bijankhan corpus (Ghayoomi, 2012), which covers all kinds of Farsi words and exceptions, were chosen. Then the words were classified based on their length, and finally the parameters of PHMM were estimated. Initial state probability vector consists of three elements: the first element refers to prefix and is estimated by counting the numbers of words which begin with prefix, the second element refers to stem and is estimated by counting the number of words which begin with stems, and the third element is zero as described before. The transition matrix is a three by three matrix, and the observation matrix is a three by thirty three matrix in PHMM (three is the number of states, and thirty three are the number of Farsi alphabet plus space character). These two matrixes are estimated based on BijanKhan corpus. To optimize the values of these parameters, expectation-maximation algorithm (EM) is used. EM algorithm computes maximum likelihood estimation iteratively when the data is incomplete (Li, Parizeau, & Plamondon, 2000). One third of BijanKhan corpus words are used to train the parameters of Persian Hidden Markov Model stemmer (Kato, Joga, Rittcher, & Blake, 2002). After training all models, the most probable path that produces a word is generated by viterbi algorithm (Rose Y, Shu, & Marc, 2003). In fact, in our model, the HMM starts from a state that has non-zero probability such as prefix or stem. It moves in states by transition distribution probability and create a symbol at each state. The viterbi algorithm finds the most probable path of producing the characters. The analysis of this path helps us find the stem, i.e. the characters before the stem are prefixes and the ones after stem are suffixes. The stemmer keeps only stem states and removes prefixes and suffixes. But there are some exceptions in Farsi, for example, the stem of some words like "پرندگان" means birds is

"پرنده", while our stemmer removes the suffix "گان", but doesn't add "ه" to the end of the word, because the base of PHMM is to remove prefixes and suffixes, not to add anything. So, we collected the exceptions of Farsi language by searching in Bijankhan corpus and Hamshahri test collection words (AleAhmad, Amiri, Darrudi, Rahgozar, & Oroumchian, 2009), and kept them in a database with their stems. This database also consists of Mokassar words. At first, a word is searched in a dictionary, if it exists there, the stemmer extracts the stem of it from the dictionary, if not, the PHMM stemmer will find its stem. Figure2 shows the steps of PHMM.

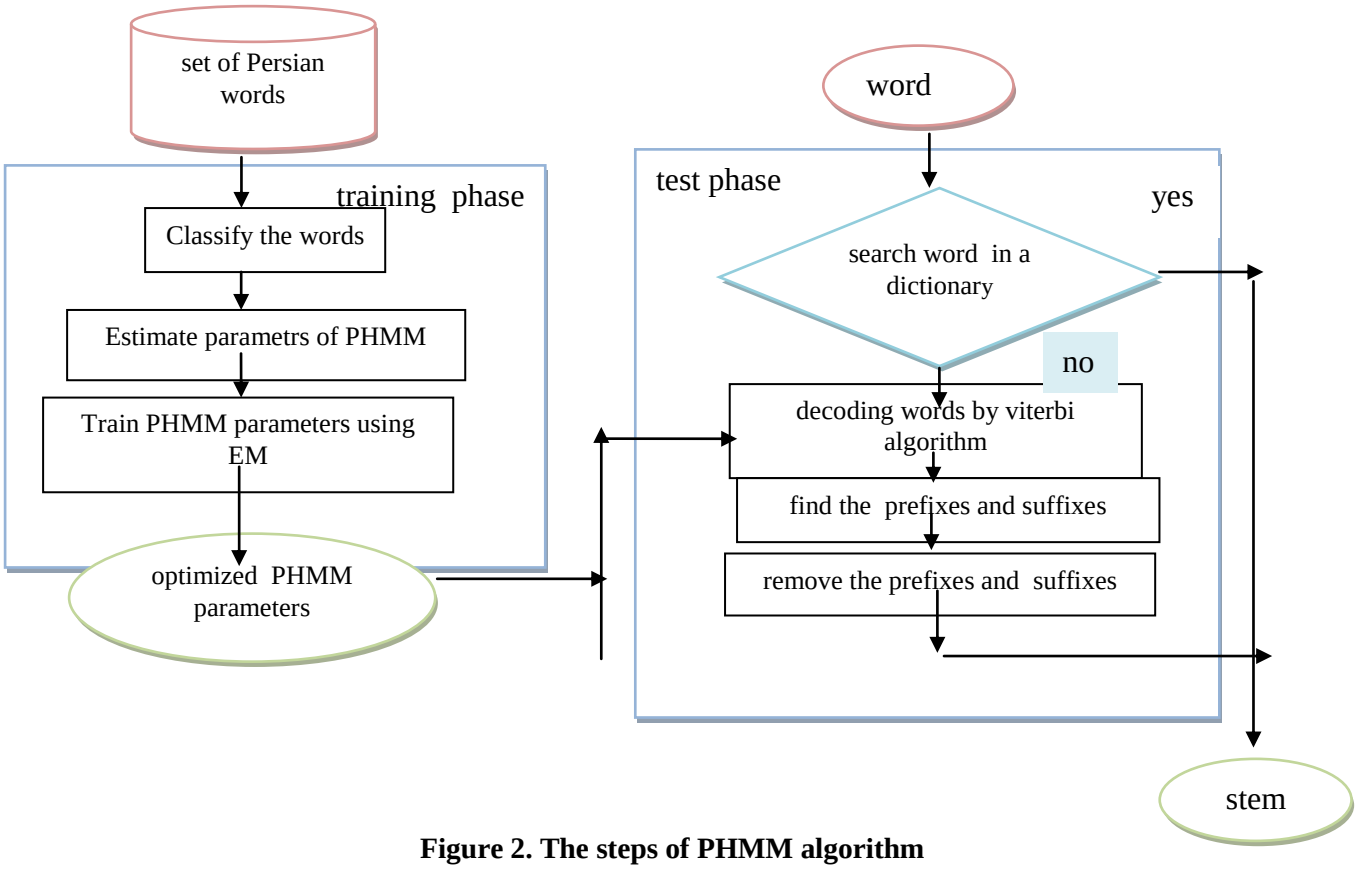

Figure 2. The steps of PHMM algorithm

Table 1 shows some words and their correct stems which are generated by PHMM stemmer.

Table1

Some of Bijankhan corpus words and their correct stem and stems by PHMM

words	Correct stem	Stem of PHMM
ژنرال (jeneral)	ژنرال (jeneral)	ژنرال (jeneral)
صاحبدلي (sahebdeli)	صاحبدل (sahebdel)	صاحبدل (sahebdel)
ضخيم (zakhim)	ضخيم (zakhim)	ضخ (zakh)

words	Correct stem	Stem of PHMM
زاهدان (zahedan)	زاهدان (zahedan)	زاهدان (zahedan)
فیلم های (film-haaye)	فیلم (film)	فیلم (film)
قالب ها (ghaaleb-ha)	قالب (ghaaleb)	قالب (ghaaleb)
کتاب (ketaab)	کتاب (ketaab)	کتاب (ketaab)
کتاب هایم (ketaab-haayam)	کتاب (ketaab)	کتاب (ketaab)
کنکور (konkur)	کنکور (konkur)	کنکور (konkur)
مبتلایان (mobtalayaan)	مبتلا (mobtalaa)	مبتل (mobtal)
نهنگ ها (nahang-haa)	نهنگ (nahang)	هنگ (hang)

Experiment and Results

To evaluate the accuracy of stemmers, the following formula can be utilized (Rahimtoroghi, *et al.*, 2010) :

$$\text{accuracy}= \frac{number\ of\ correct\ stems}{number\ of\ words} \quad (8)$$

PER-Tree-Bank words (Ghayoomi, 2012) and their stems were used to test our new algorithm which was then compared with Farsi stemmer1 (Taghva, *et al.*, 2005) and Farsi stemmer2 (Dianati, Sadrodini, & Taghizade, 2014). The results obtained from these stemmers are shown in Table2:

Table2

Results of PHMM stemmer in Per-Tree-Bank

	Farsi stemmer1	Farsi stemmer2	PHMM
Numbers of words	500	500	500
Number of correct stem	322	364	394
Accuracy	64%	73%	79%

The results indicated that the accuracy of PHMM stemmer is higher than Farsi stemmer1 and Farsi stemmer2. This is because our newly proposed method solves most of exception problems. Our newly proposed method was also tested in Bijankhan distinct words (Rose Y, *et al.*, 2003) which has 76707 words and it is much larger than Per-Tree-Bank. PHMM found 71% of stems correctly. To test the stemmer in information retrieval system, an information retrieval system and a test collection are needed. Test collection consists of sets of texts, queries and judgments that show which texts are related to each query (Aslam, Pavlu, & Yilmaz, 2006; Bijankhan, & Moradzadeh, 2004; Jalali, Moini, & Alaee Arani, 2015). To

evaluate the proposed stemmer in this study, Indri search engine (Metzler, Strohman, Turtle, & Croft, 2004), and Hamshahri test collection (AleAhmad, *et al.*, 2009) were used. The Characteristics of Hamshahri test collection are shown in Table 3.

Table3
Hamshahri test collection characteristics

File size	1400 MB
File format	XML
Number of texts	318000
Number of queries	50
Number of distinct terms	599759

There are some metrics to measure the performance of ranked information retrieval system which are described below:

- **Precision:** is the fraction of retrieved documents which are relevant to user's query. As shown in formula 9, n is the number of retrieved documents and d_i is the document which is related to the query (Keyvanpour, & Tavoli, 2013).

$$\text{precision}=\frac{\sum_{i=1}^{n} di}{n} \quad (9)$$

Recall: is the ability to find all relevant documents. As shown in formula 10, R is the number of relevant documents and d_i is the document which is related to the query (Keyvanpour, & Tavoli, 2013).

$$\frac{\sum_{i=1}^{n} di}{R}=\text{Recall} \quad (10)$$

Mean average precision: is the arithmetic mean of precision for the top k documents. In formula 11, AP is the average precision and Q is the number of queries. (Keyvanpour, & Tavoli, 2012).

$$\text{MAP}=\frac{\sum_{q=1}^{Q} AP(q)}{Q} \quad (11)$$

We tested our algorithm in indri search engine (Taghva, *et al.*, 2005) one time without stemming and the other time with PHMM stemmer and then compared the results with another stemmer (Rahimtoroghi, *et al.*, 2010). The results are shown in Table 4.

	IR system without stemmer	IR system with PHMM stemmer	IR system with Farsi stemmer3	Improvement IR with PHMM to IR without stemmer	Improvement IR with PHMM to IR with Farsi stemmer 3
Map	0.4031	0.441	0.4224	8.6%	4.21%
Recall	0.8592	0.8925	0.8656	3.73%	3%
Precision at 5	0.6620	0.7135	0.6940	7.2%	2.73%
Precision at 10	0.6280	0.6672	0.6510	5.9%	2.4%

As the table shows, in comparison of IR system with proposed stemmer and IR system without stemmer the mean average precision increased 8.6%, recall improved 3.73%, precision at 5 improved 7.2%, and precision at 10 improved 5.9%. The results of comparison of IR system with PHMM stemmer and IR system with Farsi stemmer3 show 4.21% improvement in map, 3% improvement in recall, 2.73% improvement in precision at 5 and 2.4% improvement in precision at 10. The average precision for queries of Hamshahri test collection were examined. Studying the results help us find out the strength and weakness of the proposed algorithm. PHMM stemmer finds suffix "ها"("ha"), " های" ("haaye") and all of their forms very well. It also does well in most of exceptions of Farsi words. However, it has some weaknesses. For example, it recognizes "ین "("yin") which is part of a word as a suffix by mistake, so it decreases the precision of IR system. PHMM stemmer can't find the prefix of the verbs, but as most of the users usually don't use verbs in their query, this problem can be ignored.

Precision-recall diagram is used to compare the performance of various Information retrieval systems (Keyvanpour, & Tavoli, 2012). We used this diagram for Information retrieval system with PHMM stemmer, without stemmer and with Farsi stemmer3. As Figure 3 shows, the graph of system with PHMM stemmer is always upper than the others, so PHMM stemmer improves performance of the system.

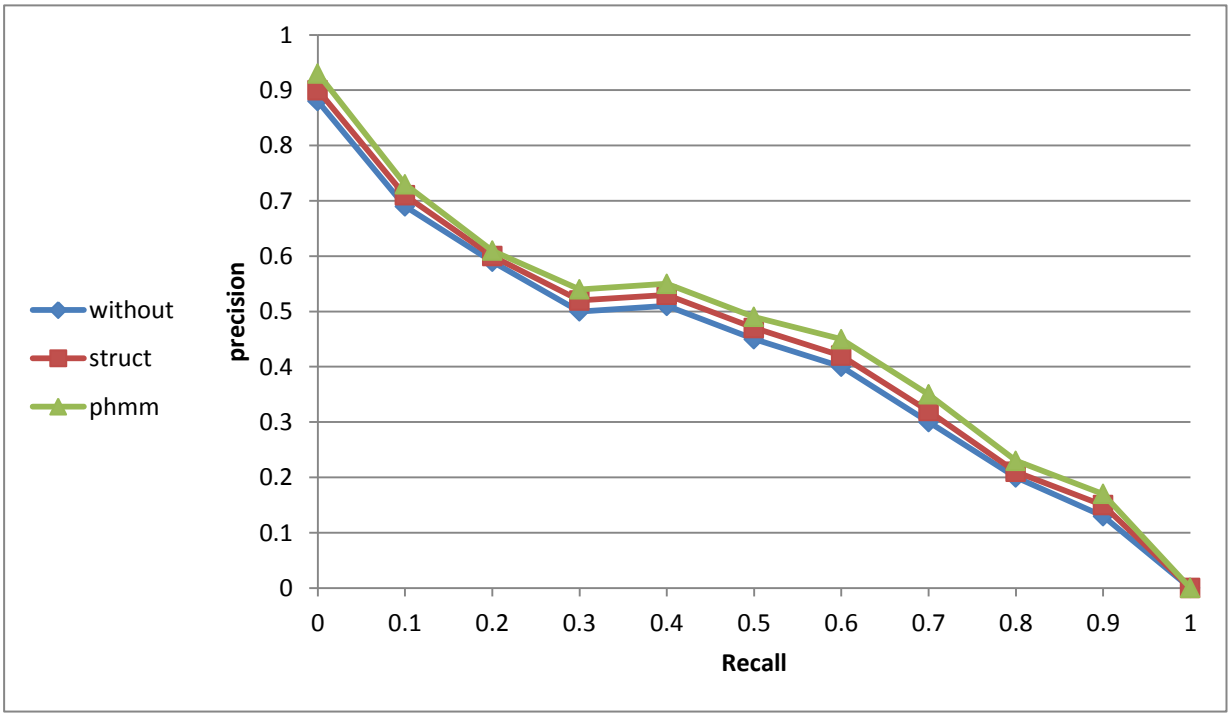

Figure3: R-precision graph of IR system without stemmer, with PHMM stemmer and with Farsi stemmer3

PHMM also decreases the size of indexed file about 6% and it also increases 5% of speed of the system.

Conclusion and Future Work

In this paper a statistical Persian stemmer based on Hidden Markov Model was designed

and tested in Per-Tree-Bank and Bijankhan corpus to evaluate its accuracy. This stemmer was also examined with Hamshahri test collection in Indri search engine. The results showed improvement in mean average precision. PHMM decreased the indexed file and increased thevspeed of Information retrieval system. The advantages of the proposed method is its independence to the language; moreover, it doesn't need to use morphological rules.

In future we can test this stemmer in Arabic text collection, we can also make a good model for the verbs.

References

AleAhmad, A., Amiri, H., Darrudi, E., Rahgozar, M., & Oroumchian, F. (2009). Hamshahri: A standard Persian text collection. Knowledge-Based Systems, 22(5), 382-387.

Aslam, J. A., Pavlu, V., & Yilmaz, E. (2006, August). A statistical method for system evaluation using incomplete judgments. In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 541-548). ACM.

Bijankhan, M., & Moradzadeh, sh. (2004). Homographs in Persian Morphology. In Proceedings of the First Workshop on Persian Language and Computers, Tehran University, Iran. May 25-26.

Dianati, M., Sadrodini, M.H., & Taghizadeh, A.H. (2014). An independent of language method to stem the persian words based on similarity measure,11th conference of Iranain Intelligent systems.

Estahbanati, S., Javidan, R. (2011). A New Stemmer for Farsi Language. Computer Science and Software Engineering(CSSE), CSI international Symposium on, 3(1), 25-29.

Estahbanati, S., Javidan, R., & Nikkhah, M. (2011). A New Multi-Phase Algorithm for Stemming in Farsi Language Based on Morphology. International Journal of Computer Theory and Engineering, 3(5), 15-23.

Ghahramani, Z. (2002). An introduction to hidden Markov models and Bayesian networks. Journal of pattern Recognition and Artificial intelligence, 15(1), 9 – 42.

Ghayoomi, M. (2012). Bootstrapping the Development of an HPSG-based Treebank for Persia. In Linguistic Issues in Language Technology, 7(1) , 1-13.

Jadidinejad, A.H., Mahmoudi, F., & Dehdari, J. (2010). Evaluation of Perstem: A Simple and Efficient Stemming Algorithm for Persian. CLEF 2009 Workshop, Part I, LNCS 6241, 98–101.

Jalali, Z.S., Moini, M. R., & Alaee Arani, M.(2015). Structural and Functional Analysis of Lexical Bundles in Medical Research Articles: A Corpus-Based Study. International Journal of Information Science and Management, 13(1), 51-69.

Kato, J., Joga, S., Rittcher, J., & Blake, A. (2002). An HMM-Based Segmentation Method for Traffic Monitoring Movies. IEEE Transactions on Pattern Analysis and Machine intelligence, 24(9), 1291-1296.

Krovetz, R. (1993). Viewing morphology as an inference process, in R. Korfhage et al., Proc. 16th ACM SIGIR Conference, Pittsburgh, 191-202.

Keyvanpour, M., & Tavoli, R. (2012). Feature weighting for improving document image retrieval system performance. International Journal of Computer Science Issues, 9(3) , 125-130.

Keyvanpour, M., & Tavoli, R. (2013). Document image retrieval: Algorithms, analysis and promising directions. International Journal of Software Engineering and Its Applications, 7(1), 93-106.

Li, X., Parizeau, M., & Plamondon, R. (2000). Training hidden Markov models with multiple observations- a combinatorial method. IEEE Transactions on Pattern Analysis and Machine Intelligence. 22(4), 371-377.

Mahdavi, M. A. (2015). Building a Syllabic Analyzer for Persian Using Finite State Transducers. International Journal of Information Science and Management, 13(1), 39-50.

Mehrad, J., & Naseri, M. (2010). The Islamic World Science Citation Center: A New Scientometrics System for Evaluating Research Performance in OIC Region. International Journal of Information Science and Management, 8 (2), 1-10.

Mehrad, J., & Berenjian, S. R. (2011). Providing a Persian Language Singular-Stemmer System (RICeST Stemmer). International journal of science and Management, 9(2), 13-22.

Mehrad, J., & Koleini, S. (2007). Using SOM Neural Network in Text Information Retrieva. Iranian Journal of Information Science and Technology, 5(1), 53-64.

Melucci, M., & Orio, N. (2003). A Novel Method for Stemmer Generation Based on Hidden Markov Models. . In Proceedings of Conference on Information and Knowledge Management (CIKM03), pages 131-138, New Orleans, LA, November 2003. ACM Press.

Metzler, D., Strohman, T., Turtle, H., & Croft,W. B (2004). Indri at TREC 2004: Terabyte Track. To appear in the Online Proceedings of 2004 Text REtrieval Conference.

Mohammad Nasiri,M., Sheikh Esmaeili, K., & Abolhassani, H.(2006). A statistical stemmer for Persian language. In 11thInt, CSI computer conf., Tehran, CSICC 2006, Iran.

Mokhtaripour, A., & Jahanpour, S. (2006). Introduction to a new Farsi stemmer. International Conference on Information and Knowledge Management - CIKM, 2006, pp. 826-827.

Momenipour Moghadam, F., & Keyvanpour, M. (2013). Analytical Study of Various Information Retrieval Models Based on Mathematical Approaches. Journal of Next Generation Information Technology (JNIT). 4(5), 63-73.

Rabiner, L.R. (1983). A tutorial on hidden Markov models and selected applications in speech recognition. Proc. IEEE, 77(2), 257–286.

Rahimtoroghi, E., Faili, H., & Shakery, A. (2010). A Structural Rule-based Stemmer for Persian. 5th International Symposium on Telecommunications.

Rose Y, Sh., Shu, L., & Marc P. C. (2003). Two Decoding Algorithms for Tailbiting Codes. IEEE Transactions on Communications, 51(10), 1358-1365.

Sharifloo, A.A., & Shamsfard, M. (2008). A Bottom up Approach to Persian stemming. Proceedings of the third joint conference on Natural language processing, 2 , 583-588.

Song, L., Boots, B., Sajid, S., Gordon, G., & Smola, A. (2010). Hilbert space embeddings of hidden Markov models, In Proceedings of the 27th International Conference on Machine Learning.

Taghva, K., Beckley, R., & Sadeh, M. (2005). A stemming algorithm for the Farsi language. International Conference on Information Technology Coding and Computing ITCC05. IEEE, 1, 158–162.