Journal of Artificial Intelligence Research 56 (2016) 61-87 Submitted 09/15; published 05/16

Automatic Wordnet Development for Low-Resource Languages using Cross-Lingual WSD

Nasrin Taghizadeh NSR.TAGHIZADEH@UT.AC.IR
School of Electrical and Computer Engineering
College of Engineering, University of Tehran, Tehran, Iran

Hesham Faili HFAILI@UT.AC.IR
School of Electrical and Computer Engineering
College of Engineering, University of Tehran, Tehran, Iran

Abstract

Wordnets are an effective resource for natural language processing and information retrieval, especially for semantic processing and meaning related tasks. So far, wordnets have been constructed for many languages. However, the automatic development of wordnets for low-resource languages has not been well studied. In this paper, an Expectation-Maximization algorithm is used to create high quality and large scale wordnets for poor-resource languages. The proposed method benefits from possessing cross-lingual word sense disambiguation and develops a wordnet by only using a bi-lingual dictionary and a mono-lingual corpus. The proposed method has been executed with Persian language and the resulting wordnet has been evaluated through several experiments. The results show that the induced wordnet has a precision score of 90% and a recall score of 35%.

1. Introduction

One of the most important projects in natural language processing over the years has been the construction of an English wordnet (WordNet) at Princeton University under the direction of George A. Miller (1995). WordNet consists of a lexical database, in which English words are grouped into sets of cognitive synonyms called synsets. The effectiveness of WordNet in a wide range of language technology applications inspired many researchers to create wordnets for other languages. The first attempts at this led to the construction of EuroWordNet (Vossen, 1998) and BalkaNet (Tufis, Cristea, & Stamou, 2004). EuroWordNet deals with European languages such as English, Dutch, German, French, Spanish, Italian, Czech and Estonian; while BalkaNet covers languages from the Balkan zone. To interconnect wordnets of different languages, EuroWordNet links synsets of each language to an interlingual index (ILI). The ILI allows it to find equivalent synsets across all languages connected to the ILI.

Although the first wordnet was created manually, several automatic and semi-automatic techniques have been used for developing the other wordnets. These methods are usually divided into merge and expansion approaches (Fellbaum & Vossen, 2012; Oliver & Climent, 2012; Erjavec & Fišer, 2006). However, there are methods that combine the merge and expansion models and benefit from the advantages of both approaches (Prabhu, Desai, Redkar, Prabhugaonkar, Nagvenkar, & Karmali, 2012; Apidianaki & Sagot, 2014). With the merge approach, a small wordnet is created manually, which contains high-level and basic concepts. Next, this small wordnet is developed using automatic and semi-automatic techniques. In this process, mono-lingual resources and language-specific properties are employed. Wordnets created in this manner later are mapped onto either the WordNet or ILI. When using the expansion approach, a multilingual wordnet is constructed by translating words inside the synsets of the WordNet (or other existing wordnets) into the target language using multi-lingual resources. Therefore the structure of the original wordnet is preserved and the words are translated.

Among the different methods proposed for wordnet construction, few of them are applicable to low-resource languages. Methods that follow the merge approach are labour-intensive and time-consuming. Moreover, they need to have vast knowledge about the language and also require many resources, which is the main obstacle of low-resource languages - so it makes this approach inapplicable for them in practice. On the other hand, methods that follow the expansion approach usually adopt WordNet structure and find the correct translation of the associated words with the WordNet synsets in the target language. In this process, multilingual resources such as comparable corpora (Kaji & Watanabe, 2006), parallel corpora (Oliver & Climent, 2012; Kazakov & Shahid, 2009; Fišer, 2009; Diab, 2004), thesaurus (Gunawan & Saputra, 2010), machine translators (Saveski & Trajkovski, 2010) and multiple bi-lingual machine readable dictionaries (Atserias, Climent, Farreres, Rigau, & Rodríguez, 2000; Patanakul & Charnyote, 2005; Bond, Isahara, Kanzaki, & Uchimoto, 2008; Lam, Al Tarouti, & Kalita, 2014) are used, which causes a bottleneck for low-resource languages.

Taking a deeper look at the expansion-based methods, each synset from the WordNet is kept and words associated with it are translated into the target language. A bi-lingual dictionary is usually employed and English words inside the WordNet synsets are translated. Since dictionaries do not translate word sense to word sense, but rather word to word, translations are ambiguous and should be disambiguated. Looking more carefully, after translating English words inside a WordNet synset, a set of candidate words in the target language is obtained; some of these are equivalent to the other senses of the English words and should thus be omitted. Methods that following the expansion approach rank the candidate words and omit low-rated ones from the candidate sets. If the task of scoring candidate words for the WordNet synsets is considered to be an optimization problem, (sub)-optimal values can be found using algorithms such as Expectation-Maximization (Montazery & Faili, 2011). The proposed method is an extension of this work for low-resource languages.

In this paper, the problem of automatically constructing large scale and high quality wordnets for low-resource languages is studied. Between the two major approaches, merge and expansion, the first one is not suitable; because it requires vast knowledge about the target language and also many language resources. So the preferred approach is to utilize wordnets in other languages by adopting their structure and translating their content. Finding the correct senses of the target language words is an AI-complete problem (Mallery, 1988), that is, by analogy to the NP-completeness in the complexity theory, it is a problem whose difficulty is equivalent to solving the central problems of AI (Navigli, 2009). In this paper, an iterative optimization method based on cross-lingual WSD is proposed to find the local optimum of the problem in a reasonable time. The main idea is to iteratively improve the estimation of the probability of selecting WordNet synsets for the words of the target language. Additionally, the proposed method needs few resources and so it is suitable for poor-resource languages. To investigate the performance of the proposed method, Persian has been selected as a poor-resource language and the resulting wordnet is examined through conducting several experiments.

The roadmap of the paper is as follows: Section 2 presents related works; Section 3 explains the wordnet construction problem and the proposed formulation; Section 4 presents a case study of the Persian language and error analysis; and conclusions are given and future works suggested in the last section, Section 5.

2. Related Work

In this section, some automatic methods for constructing wordnets are reviewed that are based on the expansion approach. The main stage of the expansion-based methods is finding a set of words that lexicalizes the concept captured by a given synset of an existing wordnet in another language. All candidate words are usually extracted by a dictionary and a scoring system is utilized to find the correct words.

In the work of Kaji and Watanabe (2006), the gloss information in WordNet has been used for the automatic construction of a Japanese wordnet. Given an English synset, it calculates a score for each of its Japanese translation candidates according to the gloss appended to the synset. The score is defined as the sum of correlations between the translation candidates and the associated words that appear in the gloss. A pair of words are deemed "associated" if the amount of mutual information between them is above a predefined threshold. Since the availability of bi-lingual corpora is limited, an iterative approach has been proposed for calculating pair-wise correlations.

Another study on creating a wordnet by automatically expanding WordNet describes a Romanian wordnet. In the work of Barbu and Barbu Mititelu (2005), in order to identify the Romanian words corresponding to a WordNet synset, several heuristics have been proposed. According to the first heuristic, words related to a synset share a common meaning. Therefore, the intersection of translations of words associated with the WordNet synsets is considered. The second heuristic states that a synset and its hypernym share the same meaning. Therefore, the intersection of word translations from a given WordNet synset and its hypernym is selected as a Romanian synset. According to the third heuristic, those translations that have the same domain label are selected for a given WordNet synset. By the fourth heuristic, a Romanian word is selected if English translations of words based on its definition have maximum similarity with words in the gloss of the given synset.

In the research conducted by Patanakul and Charnyote (2005), a semi-automatic expanding approach has been presented to construct a Thai wordnet. Candidates for links between Thai words and WordNet synsets have been derived from WordNet and its translations. To rank these links, 13 criteria are used that have been categorized into three groups: monosemic, polysemic, and structural criteria. Monosemic criteria focus on English words that have only one meaning and assume that such English words have only one synset in the WordNet. Polysemic criteria focus on English words that have multiple meanings, and believe that such English words have multiple synsets in the WordNet. Structural criteria focus on structural relations among synsets with respect to the wordnet 1.7.

Another idea for creating wordnet is to use a word-aligned parallel corpus with n languages, annotate each word with a lexical sense tag that consists of the n-tuple of aligned words. As a result, all occurrences of a given word in the text for language L are considered to have the same sense, provided they are tagged with the same multi-lingual synset. However, this kind of corpus is not easily available in most languages. In the research, which was conducted by Oliver and Climent (2012), two strategies for the automatic construction of these corpora are proposed: (i) by the machine translation of sense-tagged corpora, and (ii) by the automatic sense tagging of bi-lingual word-aligned corpora. The results for Spanish language showed that the first strategy works better than the second. This suggests that lexical selection errors made by the machine translation systems are less important than the sense tagging errors.

In the BabelNet project, which was undertaken by Navigli and Ponzetto (2010, 2012a), a very large multi-lingual semantic network was constructed. In this project, original wordnet was used as its lexicographic resource as well as Wikipedia pages in different languages for its encyclopedic knowledge. First a mapping between the English Wikipedia pages and the synsets in the original wordnet was established. Given a Wikipedia page w and its mapping, a Babel synset was created using the wordnet synset s, page w, all inter-language links, which are translation of w to the other languages. In this project, the coverage of the resulting network has been analyzed by comparing it with the gold-standard wordnets in terms of synset coverage, word coverage, and synset extra coverage. The results show that the synset coverage varies for different languages from 52% for Italian to 86% for French.

In the work of Bond and Foster (2013), an open multi-lingual wordnet for more than eighty languages was developed. In this project, a common interface for accessing multiple wordnets was created through gathering existing freely available wordnets of different languages and automatically linking them to the WordNet. Next, the wordnets were extended using the Unicode Common Locale Data Repository (UCLDR) and Wiktionary. To rank candidate links between WordNet synsets and Wiktionary, several similarity measures were employed. The results show that the precision score was 85%-99% when measured on sense.

An Arabic wordnet was created that follows the EuroWordNet methodology of manually encoding a set of base concepts while maximizing compatibility across Arabic and English wordnets (Black, Elkateb, & Vossen, 2006; Elkateb, Black, Rodríguez, Alkhalifa, Vossen, Pease, & Fellbaum, 2006). Next, in the project, which was performed by Rodrquez et al. (2008), a machine learning algorithm was employed for extending the Arabic wordnet and augmenting formal specification to the senses of its synsets. In order to associate Arabic words with the WordNet synsets, a Bayesian network with four layers was proposed. Four layers respectively represent: Arabic words; the corresponding English translation of these Arabic words in the first layer; all the synsets of the English words in the second layer; and other WordNet synsets linked to the synsets in layer three. A set of candidates word-synset is built with pairs <x, y>, where x is an Arabic word and y is a WordNet synset in the third layer of the Bayesian network that has a non-null probability and so there is a path from x to y. The score of each link is calculated with the posterior probability of y, given the evidence provided by the network. Only the tuples that score over a threshold are selected for inclusion in the final set of candidates word-synset. The best results of the method proposed in this study noted a score of 71% precision.

In the work of Boudabous et al. (2013), an Arabic wordnet was enriched via adding semantic relations between synsets. The method consisted of two main phases; the first phase consisted of defining morpho-lexical patterns using a study corpora extracted from the Arabic Wikipedia. The second phase consisted of using morpho-lexical patterns, defined in the previous phase, in order to extract new semantic relations from the Arabic Wikipedia. Extracted relations were validated, then added to the Arabic wordnet data base.

Piasecki et al. (2011) proposed an algorithm for automatically expanding the Polish wordnet. This method uses heterogeneous knowledge sources, which are extracted from a large corpus, and combines them based on a weighted voting scheme. This method extracts potential instances of lexicon-semantic relations from a corpus and measures the semantic similarity of lexical units. It analyzes the effect of using different knowledge resources on the performance of the algorithm. Due to the high accuracy of the results, this approach can be said to be a good basis for semi-automatic methods of constructing wordnets using human knowledge to correct the output of the automatic approaches.

Lam et al. (2014) proposed an automatic method for constructing wordnet synsets that uses the publicly available wordnets, a machine translator and bi-lingual dictionaries. For this purpose, each synset of an existing wordnet is translated into the target language, then a ranking method is applied to the resulting translation candidates to find the best translations. To generate candidate synsets, three approaches were proposed; The first one directly translates synsets in WordNet into the target language. The second one uses intermediate wordnets to handle ambiguities in synset translations. In the case of dictionaries being available, in addition to the wordnets in the intermediate languages, a third approach can be used. The experimental results showed that the resulting wordnets have a coverage of 19%, 65%, 37%, 21% and 83% for Karbi, Arabic, Assamese, Dimasa and Vietnamese languages, respectively.

In the project, which was conducted by Hasanuzzaman et al. (2014), a method for constructing a Tempo-wordnet was suggested. According to this method, the WordNet was augmented with temporal information by following a two-step process: in the first step, synsets of the WordNet are classified as *atemporal* or *temporal*. Next, all synsets are associated with *past*, *present* and *future* probabilities. The obtained Tempo-wordnet can be used in time-related applications.

In the work of Shamsfard (2008), a semi-automated method was proposed for developing a Persian lexical ontology called FarsNet. About 1,500 verbs and 1,500 nouns were gathered manually to make the wordnet's core. After that, two heuristics and a Word Sense Disambiguation (WSD) method were used to find the most likely related Persian synsets. A practical evaluation of the proposed automatic method used in this studt shows a score of 70% correctness and covers about 6,500 entries on WordNet. The extension of this work (Shamsfard, Hesabi, Fadaei, Mansoory, Famian, Bagherbeigi, Fekri, Monshizadeh, & Assi, 2010a), is known as being the first published Persian wordnet, FarsNet, which contains about 18,000 Persian words and covers about 6,500 WordNet synsets.

In the research, which was performed by Montazery and Faili (2010), an automatic approach for Persian wordnet construction based on the WordNet has been introduced. The proposed approach uses two mono-lingual corpora for English and Persian, and a bilingual dictionary in order to construct mapping between WordNet synsets and Persian words using two different methods; some links were selected directly by using heuristics that recognize these links as unambiguous. Other types of links are ambiguous, in which a scoring method is used to select the appropriate synset. The practical evaluation of the links for 500 randomly selected Persian words shows about 76.4% quality in terms of accuracy.

By augmenting the Persian wordnet with unambiguous words, the total accuracy of the automatically extracted Persian wordnet becomes 82.6%.

3. Iterative Method for Wordnet Construction

To construct a multi-lingual wordnet, several methods have been presented; however, few of them have paid attention to low-resource languages. Creating a wordnet from scratch for such languages is a time-consuming and expensive process. Instead, new wordnets could be developed by adopting the structure of existing wordnets in other languages (usually WordNet) and translating the words associated with their synsets into the target language. One important advantage of this approach is that the resulting wordnet is aligned to the WordNet and the ILI, and thus is interesting for contrastive semantic analysis and is particularly useful in multi-lingual tasks such as multi-lingual information retrieval (Dini, Peters, Liebwald, Schweighofer, Mommers, & Voermans, 2005; Otegi, Arregi, Ansa, & Agirre, 2015) and multi-lingual semantic web (Buitelaar & Cimiano, 2014). The main assumption on which one can develop a wordnet using the expansion approach is that most of the concepts and semantic relations are common among different languages. Therefore, language-specific concepts and relations may not be covered in the resulting wordnet.

In general, and regardless of the approach taken, the main step toward constructing a complete wordnet is to generate synonym sets. In this section, an automatic method for extracting synsets for languages with limited resources is proposed. The proposed method follows the expansion approach; at the start, wordnet is initialized with WordNet synsets. For every WordNet synset s, all translations of English words inside s are extracted from bilingual dictionary and links between translation words and WordNet synsets are established. Since dictionaries translate word to word, not word sense to word sense, translations are ambiguous. Therefore, the task is to score links and find incorrect ones. We consider these scores to be the probability of selecting each candidate synset for each word in the target language.

In this paper, the task of finding correct the translation of words associated with the WordNet synsets is regarded as an optimization problem. If a sensed-tagged corpus similar to the English SemCor (Landes, Leacock, & Tengi, 1998) exists in the target language, the problem of creating wordnet is converted to the maximum likelihood estimation (MLE). The English SemCor corpus is a sense-tagged corpus created at Princeton University by the wordnet project research team. The corpus consists of a subset of the Brown Corpus and contains about 700,000 words. In SemCor all the words are POS tagged and more than 200,000 content words are sense-tagged with reference to the WordNet lexical database. Since such resources may not exist, we use a word sense disambiguation method to find correct sense of each word in a raw corpus. As shown in the research, which was conducted by Mallery (1988), WSD is an AI-complete problem whose difficulty is equivalent to solving the central problems of AI. This class of problems is analogous with NP-complete problems in complexity theory, which are classified as being the most difficult problems. The proposed idea is to use an iterative algorithm that finds the local optima of the problem with few iterations in a reasonable time. Our work can be regarded as an extension of the work which was performed by Montazery and Faili (2011). The proposed method adopts this

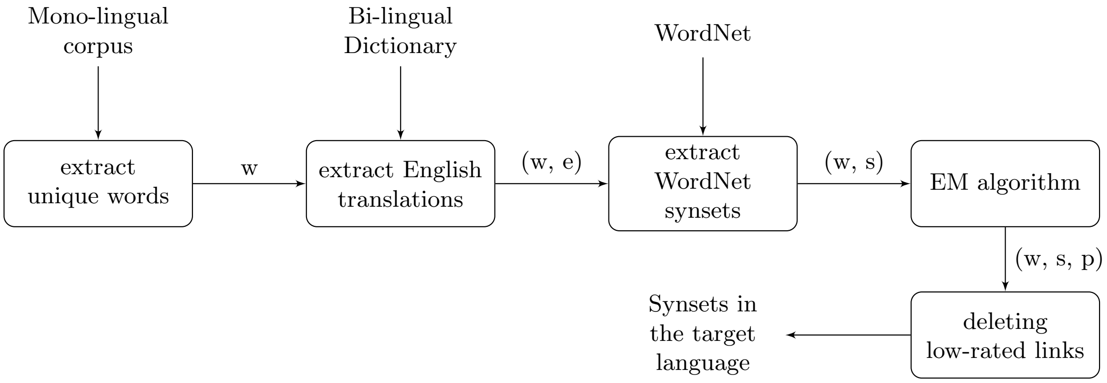


Figure 1: The overview of the proposed approach for constructing wordnet

work for low-resource languages; and our method additionally attempts to solve its major drawbacks.

The idea proposed in the work of Montazery and Faili (2011) for wordnet construction, is to use a bi-lingual dictionary as well as a raw-corpus. First, for each Farsi word in the corpus, all translations are extracted from the bi-lingual dictionary. Next, all synsets of the English translations are considered as the candidate synsets for the Farsi word. A score is calculated for each pair of Farsi words and WordNet synsets using the expectation-maximization (EM) algorithm. In the expectation step, they use a relative-based WSD method (PMI), in which the co-occurrence frequency of pairs of words in the Farsi language have been used to disambiguate words of a corpus. Experimental results showed that the precision of this method varies for different POS tags. The highest precision is shown for adjectives which is 89.7%; next for adverbs, which is 65.6%; and the lowest precision is for nouns at 61.6%.

The major drawbacks of the above method are that calculating the co-occurrence between each pair of words in the target language usually requires a large corpus, which may not be easily found in low-resource languages; this is important because the quality of the resulting wordnet highly depends on the co-occurrence values. As a result, we propose to change the expectation step of the PMI-based algorithm so that the WSD procedure can be performed without needing an additional corpus or any other language resources. Figure 1 represents an overview of the proposed method. Next, in the experimental analysis, we will re-implement this work as the baseline and compare the proposed method with it.

EM is an iterative algorithm for finding the maximum likelihood parameters of a statistical model in cases where the equations cannot be directly solved. These models typically consist of latent variables in addition to unknown parameters and known data observations. That is, either there are missing values among the data, or the model can be formulated more simply by assuming the existence of additional unobserved data points. The basic idea of the EM is as follows:

1. If we have the actual sufficient statistics for the data, we can compute the parameter values that maximize the likelihood of the data. This is just the problem of learning a probabilistic model from complete data.

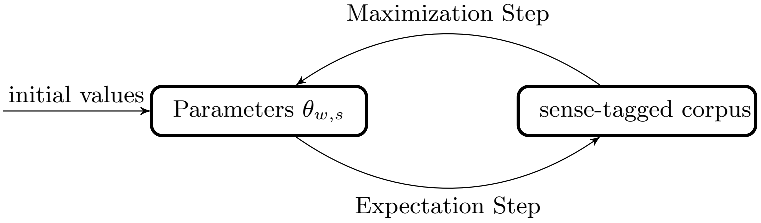


Figure 2: Expectation-Maximization algorithm for wordnet construction

2. If we actually succeed in learning the model parameters, we could then compute a probability distribution over the values of the missing attributes.

In the case of our problem, the EM algorithm should find the probability of mapping each word in the target language to each of its candidate synsets. If a candidate synset represents a correct sense for a word in the target language, it is expected that this sense occurs in a corpus containing that word. So the observed data is the words of a corpus in the target language; the unseen part of each data is the WordNet sense tag of the words.

Th EM algorithm switches between two stages: 1) finding an approximate distribution of missing data given the parameters; and 2) finding better parameters given the approximation. The first step is known as the expectation or E-step, while the second step is called the maximization or M-step. Figure 2 represents an overview of the EM algorithm used for learning words connected to the WordNet synsets. Next, details of each step in the proposed algorithm are presented.

3.1 E-Step

Similar to the work of Montazery and Faili (2011), for each word in the target language, w, and each a WordNet synset, s, $\theta_{w,s}$ is defined as the probability of choosing WordNet synset s for word w, $P(s|w)$. In other words, the number of times that word w appears in a large corpus with sense s divided by total number of appearance w. That is:

$$\forall w, s : \qquad \theta_{w,s} \in [0, 1]. \tag{1}$$

$$\forall w : \qquad \sum_s \theta_{w,s} = 1. \tag{2}$$

At this step, current values of parameters $\theta_{w,s}$ are used to label the corpus with sense tags. For each word w appearing in the corpus, an appropriate sense among the candidate WordNet synsets should be chosen. To do this task, an unsupervised cross-lingual word sense disambiguation (WSD) could be employed. WSD algorithms aim to resolve word ambiguity without the use of annotated corpora. Unsupervised WSD is a well-studied task in the literature. Among these, two categories of knowledge-based algorithms have gained popularity: overlap- and graph-based methods. The former owns its success to the simple intuition that underlies that family of algorithms, while the diffusion of the latter started growing after the development of semantic networks (Basile, Caputo, & Semeraro, 2014).

Within the graph-based framework for WSD, a graph is built from a lexical knowledge base (usually WordNet) representing all possible senses of the word sequence that is being disambiguated. Graph nodes correspond to word senses, whereas edges represent dependencies between senses. These dependencies include hypernymy, synonymy, antonymy, etc. Next, the graph structure is analyzed to determine the importance of each node. Finding the right sense for each word in the sequence amounts to identifying the most important node among the set of graph nodes representing its candidate senses. The main challenge of the graph-based WSD methods is how to create the graph, especially which dependencies should be chosen as the graph's edges, and which connectivity measure should be used to score the nodes of the graph.

In the research, which was conducted by Navigli and Lapata (2010), a comprehensive study on unsupervised graph-based WSD was conducted. They evaluated a wide range of local and global measures of graph connectivity with the aim of isolating those that are particularly suited for this task. Local measures include degree, page-rank, HITS, KPP and betweenness, whereas global measures consist of compactness, graph entropy, and edge density. Their results indicate that local measures yield a better performance than global ones. The best local measures are Degree and PageRank.

For the task of wordnet development, we adapt a graph-based WSD method as presented in work of Navigli and Lapata (2010), for the problem of the sense labelling of the corpus using the current parameters $\theta_{w,s}$. It is assumed that the true sense of each word in the corpus is determined through senses of other words in the same sentence. For every sentence of the corpus, the following procedure is executed:

- For each word w in the sentence, candidate WordNet synsets are picked, and one terminal node for each synset s in the graph is created. This set of terminal nodes is called V_w.

- For each terminal node v, a depth-first search (DFS) on the WordNet graph is performed. Every time a node $v' \in V_{w'} (w \neq w')$ along a path of length $\leq L$ is encountered, all intermediate nodes and edges on the path from v to v' are added to the graph. L is a parameter of the algorithm and usually takes small values such as 3, 4 or 5.

- Terminal nodes of the graph are scored according to their degree as follows: For node $v \in V_w$,

  $$C(v) = \frac{deg(v)}{max_{u \in V_w}(deg(u))}, \tag{3}$$

  where $deg(v)$ is the number of edges terminating in v in graph $G = (V, E)$:

  $$deg(v) = |\{(u, v) \in E : u, v \in V\}|, \tag{4}$$

Relations chosen as the graph's edges consist of all the lexical and semantic relations defined in WordNet in addition to the *gloss relation*. A pair of synsets s and s' is connected via a gloss relation if an unambiguous word $w \in s'$ occurs in the gloss of s. The word w must be unambiguous; otherwise, s should have been connected with the appropriate sense of w (Navigli & Lapata, 2010). To use gloss relation in the WSD procedure, sense disambiguated glosses of the WordNet are utilized (Semantically Tagged glosses, 2016), in which word forms from the glosses in WordNet's synsets are manually linked to the context-appropriate sense in WordNet. Therefore, gloss relation is established between s and s', if s appears as the correct sense of any word in the gloss of the s'.

The time complexity of calculating a degree measure is less than PageRank, and its performance has been shown to be better; so in the last step of the WSD procedure, a degree measure is preferred for scoring nodes of the graph. To illustrate the steps of the WSD procedure, we provide an example in the next section.

3.1.1 WSD of a Persian Sentence

In order to better understand WSD procedure, an example is presented. Consider the following Persian sentence which means "Workers with thirty years of service become retired."

$$\underbrace{\text{.}}_{punc}\ \underbrace{\text{میشوند}}_{verb}\ \underbrace{\text{بازنشسته}}_{adj}\ \underbrace{\text{خدمت}}_{noun}\ \underbrace{\text{سابقه}}_{noun}\ \underbrace{\text{سال}}_{noun}\ \underbrace{\text{سی}}_{num}\ \underbrace{\text{داشتن}}_{noun}\ \underbrace{\text{با}}_{prep}\ \underbrace{\text{کارمندان}}_{noun}$$

Preposition, number and punctuation tags are not involved in the wordnet and so are ignored. Consider the word بازنشسته/retired in the above sentence. According to the Aryanpour dictionary, this word has three translations: emeritus; pensionary; retired. According to the wordnet 3.0: the first translation has one noun synset and one adjective synset; the second one has two noun synsets; and the third one has eleven verb synsets and one adjective synset. Since this word can be a noun or an adjective in a Persian corpus, verb synsets are ignored. The definitions of the other synsets are as follows:

- {10051861} (noun.person) emeritus#1 – (a professor or minister who is retired from assigned duties)
- {01645490} (adj.all) emeritus#1 – (honorably retired from assigned duties and retaining your title along with the additional title 'emeritus' as in 'professor emeritus')
- {10414612} (noun.person) pensioner#1, pensionary#1 – (the beneficiary of a pension fund)
- {10176913} (noun.person) hireling#1, pensionary#2 – (a person who works only for money)
- {00035368} (adj.all) retired#1 – (no longer active in your work or profession)

Therefore, the candidate set for the Persian word بازنشسته/retired consists of these five synsets. In general, each of these synsets could be the correct sense in the above sentence. However, the POS tag of this word in the given sentence can come to our aid during the WSD procedure in order to filter some synsets. Indeed in the WSD procedure, only those

Table 1: Persian words and their candidate synsets.

Persian word	POS	Translations	candidate synsets	selected synset	correct
کارمند	noun	employee, worker, member	10	$worker_n^1$	✓
داشتن	noun	relieve, own, have	1	$have_n^1$	✓
سال	noun	year	4	$year_n^1$	✓
سابقه	noun	background, antecedent, history, record, service	40	$record_n^1$	✗
خدمت	noun	work, job, activity, profession	30	job_n^1	✓
بازنشسته	adjective	retired, emeritus,	2	$retired_a^1$	✓
شدن	verb	wind, grow, lapse, branch, become, be	42	$grow_v^3$	✗

synsets which have the same POS as the given POS in the sentence should be involved. Since the word بازنشسته/retired has an adjective POS in the above sentence, only adjective synsets are involved in the graph's construction. Following the above steps for the other words of the sentence leads to finding the candidate synsets of each word that should be accounted for in the WSD graph. Table 1 represents Persian words, their translations, and the number of candidate synsets regarding the POS tag of the Persian words. All of these candidate synsets represent the terminal nodes of the WSD graph. As Figure 3 shows, the candidate synsets of each Persian word of the given sentence have been grouped in a dotted box.

In the next step, a DFS algorithm is run for each terminal node on the WordNet graph with the length being at most three. Upon finding a path from one terminal node to another, all intermediate nodes and edges are added to the WSD graph. Part of the WSD graph is shown in Figure 3. Each word in this graph is associated with a POS, which is denoted with a subscript: n stands for noun, v for verb, a for adjective, and r for adverb. The superscript denotes the sense number associated with that word in WordNet 3.0. This graph has three separate components; one component for each word میشوند/become and سال/year and the other component for remaining words. This means no word in the given sentence indicates the sense of these words.

After the construction of the WSD graph, the correct sense of each Persian word should be determined. To do this, the synset with the most degree among the candidate set of each word is chosen as the correct synset for that word. Consider the word بازنشسته/retired; in the WSD graph of Figure 3, the node $retired_a^1$ has a degree of one; whereas the node $emeritus_a^1$ has a degree of zero. So the selected sense for this word is $retired_a^1$. Using the degree measure, the selected sense for each word of the given sentence is determined, which is represented in the bold box. Table 1 summarizes the steps taken in the WSD procedure of the given sentence. As the last column shows, the selected sense for all of the words is correct except for سابقه/background and شدن/become.

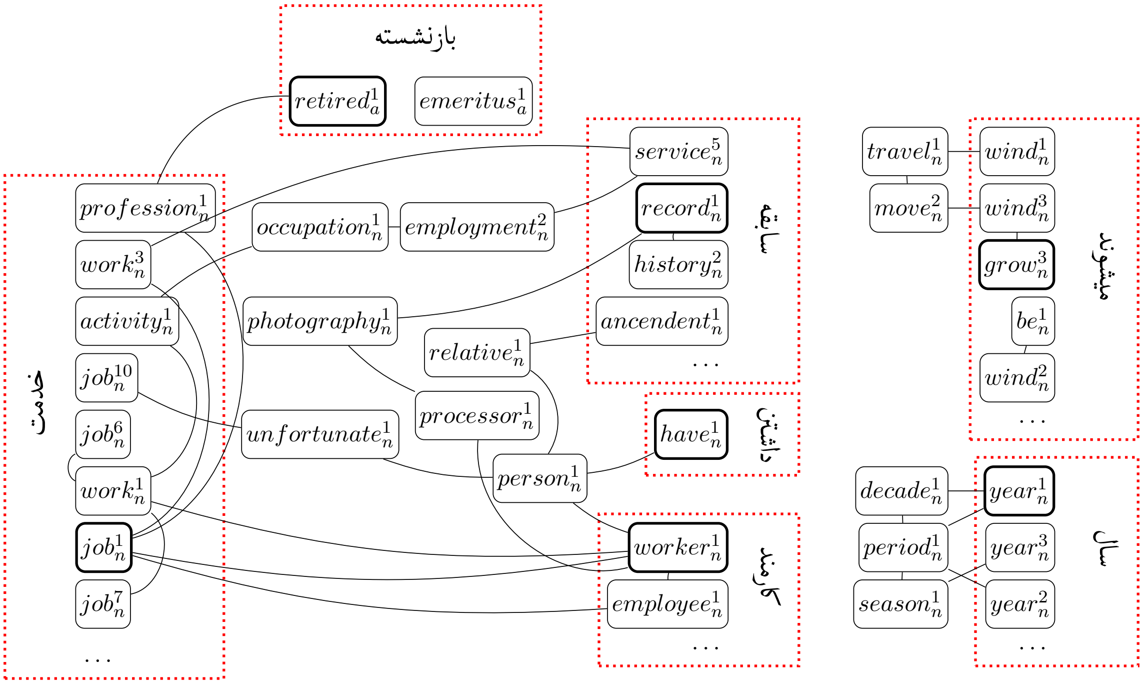


Figure 3: Part of WSD graph for the sentence کارمندان با داشتن سی سال سابقه خدمت بازنشسته میشوند.

3.2 M-Step

In the maximization step, a new estimation of the model's parameters should be calculated based on the sense-tagged corpus that resulted from the expectation step. Similar to the work of Montazery and Faili (2011), on iteration j, the new value for parameter $\theta_{w,s}$, which denotes the probability of assigning a sense tag s to the word w, is equal to averaging the conditional probability $P(s|\Theta^{j-1})$ over different occurrences of w in the corpus, where Θ^{j-1} is the set of all parameters $\theta_{w,s}$ on iteration $j-1$. In formal notation,

$$\theta^j_{w,s} = \frac{\sum^n_{\substack{i=1 \\ w_i=w,s_i=s}} P(s_i|w_1^n, \Theta^{j-1})}{N(w)}, \tag{5}$$

where $\theta^j_{w,s}$ denotes the value of $\theta_{w,s}$ on iteration j, w_1^n presents sequence of corpus words and $N(w)$ is number of occurrence of w in w_1^n.

In each iteration of the EM algorithm, the likelihood of the data given the new parameter values is at least as great as the likelihood given the old ones. So EM behaves similar to the gradient descent; at each step, it adjusts the parameter values so as to improve the likelihood of the data. It follows that EM converges to a set of parameter values that locally maximizes the likelihood.

The proposed EM method is repeated until the changes in the probability of selecting a candidate synset for a word in the target language becomes negligible. So, at the end of each iteration, the maximum change of probabilities is computed. If this value is less than t, the algorithm stops. After execution of the EM algorithm, all links with a score of below the threshold t_{remove} ($\theta_{w,s} \leq t_{remove}$) will be deleted from the wordnet. Also in each

Table 2: Assigned probabilities for word "بازنشسته/retired" per iteration.

Synset ID	Correct	Itr #0	Itr #1	Itr # 2	Itr #3	Itr #4	Itr #5
Noun:10051861	✗	0.2	0.11111	0.11111	0.11111	0.11111	0.11111
Adjective:01645490	✗	0.2	0.29885	0.08315	0	0	0
Noun:10414612	✗	0.2	0.11111	0.11111	0.11111	0.11111	0.11111
Noun:10176913	✗	0.2	0.11111	0.11111	0.11111	0.11111	0.11111
Adjective:00035368	✓	0.2	0.36781	0.58350	0.66666	0.66666	0.66666
Entropy		2.1502	1.8340	1.7880	1.7797	1.7781	1.7768

iteration, those links with a current score below t are ignored and the corresponding senses are not presented in the graph's construction and the WSD procedure. At the end, those words in the target language that are mapped onto the same synset in the WordNet make synsets of the resulting wordnet.

To better follow the process of updating probabilities of each word per iteration, an example is presented here. For demonstrating the probability adjustment in each iteration, consider again the word بازنشسته/retired. In the expectation step, all words of the corpus should be disambiguated. Next in the maximization step, the new value of the probabilities is computed. Table 2 represents the probabilities of synsets assigned to the word بازنشسته /retired in each iteration. The first and the second columns show the synset ID and the correction of synsets for the specified word, respectively. The following columns represent the probability values of the first five iterations. Values less than 0.005 were considered to be 0. This table shows that the probabilities start out uniformly; then in each iteration, the probability of correct synsets increases and the probability of incorrect synsets or those that are not frequent enough in the corpus decreases or does not change. Indeed, if the number of occurrences of word بازنشسته/retired in the corpus, which are tagged with a specific WordNet sense in iteration i are the same as the iteration $i-1$, the probability of that sense of the given word بازنشسته/retired does not change in the iteration i. If this value becomes greater, that probability increases and so if this value becomes smaller, that probability decreases. In this particular example, after five iterations, the synset achieving the highest probability is the correct synset. In iteration three, the probability of the word بازنشسته/retired being assigned to the second synset goes down to 3.9E-7, which is below the threshold. So in the next iterations, this synset is not considered in WSD procedure and its probability will be zero. The last row of the table presents the entropy value in respect to the iteration. The steady decrease in entropy indicates that in each iteration, the distinction between candidates synsets for each word becomes more clear, which leads to identification of the correct synsets. The subject of analysis of the entropy for each word per iteration is discussed later in Section 4.2.1.

4. Case Study: Persian Language

In this section, the proposed method for automatic wordnet construction is applied to Persian as a low-resource language. In the following subsections, the experimental setup and evaluation methods are described; after that, the results are presented.

4.1 Experimental Setup and Data

In this section, the required resources and setup of the experiments are explained[1]. To construct a wordnet for the Persian language, the Bijankhan Persian corpus[2] has been used. This collection has been gathered from daily news and common texts, in which all documents are categorized into different subjects such as political, cultural and so on. Bijankhan contains about ten million manually-tagged words with a tag set containing 550 fine-grained Persian POS tags (Oroumchian, Tasharofi, Amiri, Hojjat, & Raja, 2006).

Although POS tags are not explicitly used in the proposed method, to get better WSD results, one can use POS tags to prune synsets along with the other tags from the candidate set of each word as explained in Section 3.1.1. As a result, in the WSD procedure, just those synsets with the same POS tag as the word of the corpus are taken part. In WordNet, four categories of tags are included: noun, verb, adverb and adjective. Thus the words of the corpus with other tags such as pronoun and preposition are ignored.

Bijankhan is a large corpus. Most low-resource languages may not have such a large corpus. In order to evaluate the behaviour of the proposed method when the corpus size is limited, a part of the Bijankhan has been picked for training Persian wordnet. So both the PMI-based and the graph-based method have been conducted using this part. This part includes nearly 13% of the total size of the corpus. The remaining 87% has been used in the testing phase in which coverage of the wordnet over the corpus was evaluated. More details about the coverage analysis are presented in Section 4.2.4. Also, a complete analysis on the effect of the corpus size on the quality of the final wordnet is presented in Section 4.4.

Those words in the corpus that appear in their inflected forms may not be found in the dictionary. Therefore before the beginning of the proposed algorithm, a lemmatizer should be used so that different inflected forms of words are converted to their base form. For example, plural nouns should be converted to their singular form. To do this, STeP-1 tool (Shamsfard, Jafari, & Ilbeygi, 2010b) has been utilized. The STeP-1 package is a set of fundamental tools for Persian text processing that provides support for tokenization, spell checking, morphological analysis, and POS tagging.

Another required resource for the proposed method is a bi-lingual machine readable dictionary. An electronic version of the Aryanpour dictionary[3] has been used to extract the English equivalent for Bijankhan words. Also, WordNet version 3.0 has been used to extract synsets of their English equivalents.

In the WSD procedure, the context of each word is the sentence containing that word. A depth-first search in WSD has been performed up to a maximum depth of 3 similar to the work of Navigli and Ponzetto (2012b). As mentioned before in Section 3.2, if the probability of the WordNet sense s given for the word w is less than or equal to t, that sense is ignored in the WSD process of the EM algorithm. In our experiments, we have set $t = 0.005$.

[1] The source code is freely available for download at http://ece.ut.ac.ir/en/node/940

[2] See http://ece.ut.ac.ir/dbrg/bijankhan/

[3] See http://www.aryanpour.com

Table 3: Entropy values with respect to the iteration

Iteration	0	1	2	3	4	5	6
Entropy	2.15025	1.83406	1.78804	1.77978	1.77813	1.77680	1.77677

4.2 Evaluation Results

In this section, the results of the evaluation of the proposed method in various experiments are presented.

4.2.1 Convergence of the Proposed Method

The EM algorithm iterates between the expectation and maximization steps, until some criteria are satisfied. In our experiment, after each iteration, the entropy of synset probabilities per word is calculated and the average of the entropy of all the words is considered. If the changing of this value in two consecutive iterations becomes near zero, the EM algorithm stops. Formally, the entropy of a probability distribution is defined as equation 6:

$$H(w) = \sum_{s} \theta_{w,s} log(\theta_{w,s}). \tag{6}$$

Entropy is best understood as a measure of uncertainty, as entropy is larger for more random values. Indeed at first, all links for a Persian word have equal probability, and so maximum entropy is granted. After each iteration, some links sink under the threshold probability and thus the probability of the other links increases. It is expected at the final step that all incorrect links obtain a very low probability and that correct links obtain a high probability. Therefore, entropy analysis can demonstrate the behaviour of the EM method in changing probabilities. In Table 3, the entropy values per iteration are shown. At iteration 6, changing the entropy values reaches the predetermined threshold of 0.001 and the EM algorithm stops.

4.2.2 Precision and Recall of the Wordnet

The primary goal of this work is to construct a high quality wordnet for low-resource languages. After execution of the EM algorithm, the probability of assigning each candidate synset to each word in the target language is finalized. These probabilities are sorted and those links with a probability under the threshold t_{remove} should be removed from the final wordnet. The value of t_{remove} determines the size of the wordnet and affects the quality of the wordnet. So, experiments were conducted that used different values for the t_{remove} including 0.1, 0.05, 0.02, 0.01, 0.005 and 0.0.

To evaluate the resulting wordnet, we re-implemented the PMI-based method (Montazery & Faili, 2011) and compared our wordnet with it as a baseline. In all experiments, our wordnet is referred to as the "graph-based wordnet", in contrast to the "PMI-based wordnet". In the evaluation process, two data sets were used: 1) FarsNet 2) Manual judges. FarsNet is a semi-manually created wordnet in Persian, which is available in two versions; the second release of FarsNet contains more than 36,000 Persian words and phrases that are organized into about 20,000 synsets of nouns, adjectives, adverbs and verbs. FarsNet 2 has also inter-lingual relations that connect some of the Persian synsets to English ones in the Princeton wordnet 3.0.

The second data set consists of a subset of 1,750 links in the resulting wordnet, which were selected randomly and judged manually. Each link (w, s) was given to two annotators to decide if the Persian word w is semantically equal to the WordNet s. To ensure the accuracy of the judges, annotators were selected among people who are native speakers of Persian and at the same time learn English professionally. In the case of disagreement between two judges, a third annotator was asked to decide about the link. The inter-annotator agreement was 80%, which means that in 80% of judgements, the two annotators agreed. Additionally, we computed Cohen's Kappa coefficient (Cohen, 1960), for two annotators, which takes into account the amount of agreement that could be expected to occur through chance. Kappa is computed as follows:

$$\kappa = \frac{p_o - p_e}{1 - p_e}, \tag{7}$$

where p_o is the relative observed agreement among annotators, and p_e is the hypothetical probability of chance agreement. For our two annotators, the Kappa value was 0.55. In general, if the annotators are in complete agreement, then $\kappa = 1$. If no agreement between annotators other than what would be expected by chance (as given by p_e), then $\kappa \leq 0$. After carrying out the manual judgements, the precision and recall of the resulting wordnet were measured on this set.

The precision of the resulting wordnet is defined as the number of correct links in the wordnet that also exist in the test data as correct links, divided by the total number of links in the wordnet that exist in the test data. Also, the recall of the wordnet is defined as the number of correct links in the wordnet that also exist in the test set as correct links, divided by the total number of correct links in the test set. The accuracy of the wordnet is another measure, which is defined as the number of correct links in the wordnet that also exist in the test set plus the number of incorrect links in the test set that do not exist in the wordnet, divided by the total number of links in the test set. These definitions of precision, recall, and accuracy of the wordnet were also used in the BabelNet project (Navigli & Ponzetto, 2010).

Figure 4a and Figure 4b represent the precision and recall of the PMI-based method and the proposed method according to FarsNet. As shown, the precision and recall of our wordnet is better than the PMI-based method. In these figures, precision is at most 18%, which seems low for a wordnet to be considered as a reliable resource for that language. Additionally, recall is at most 49%. This is due to the lack of correct links in FarsNet. In the evaluation of the resulting wordnet according to FarsNet each link (w, s) can be placed in one of these categories:

- Persian word w does not exist in FarsNet. This link is ignored and is not counted.
- Persian word w exists in FarsNet; however no WordNet synset is given for it. This link is ignored and not counted.
- Persian word w exists in FarsNet and at least one WordNet synset is given for it. If s is one of these WordNet synsets, this link is counted as correct or else it is counted as incorrect.

The WordNet sense distinctions are too fine-grained, meaning that several WordNet synsets may be mapped onto one synset in FarsNet; while most of them are not given in FarsNet. Therefore, some correct links in our wordnet are counted as incorrect. Figure 4c shows the accuracy of the wordnets according to FarsNet, which shows that the graph-based wordnet surpasses the PMI-based wordnet.

Some reasons for low precision according to FarsNet are as follows:

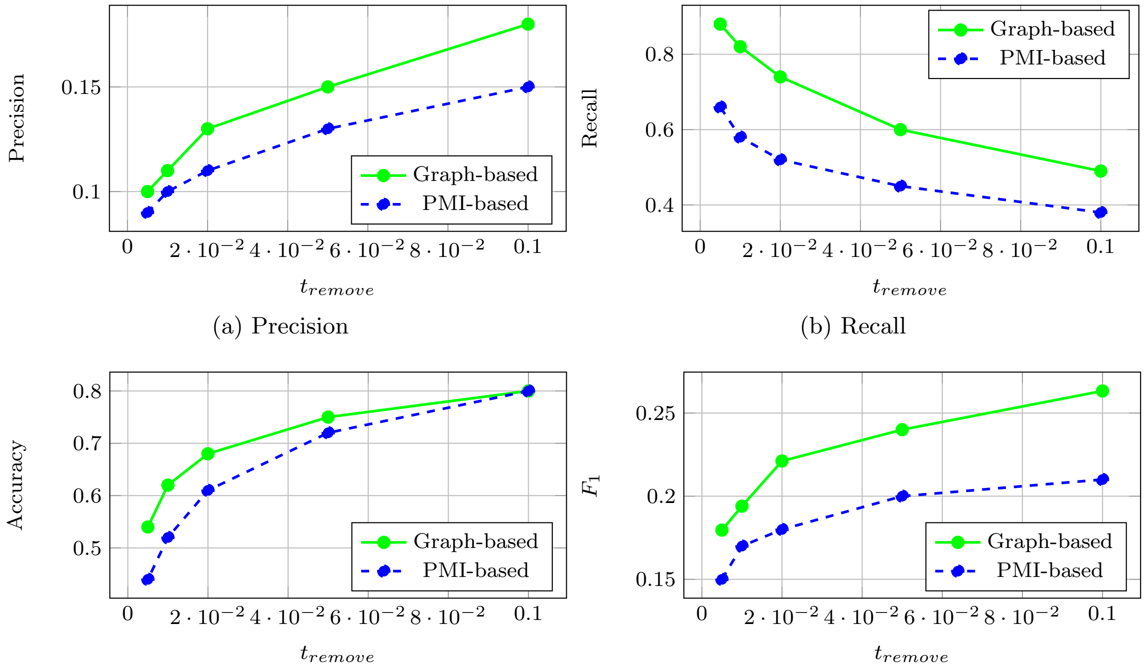
- Translations of the Persian words are inaccurate or incomplete, meaning that the correct WordNet synset according to FarsNet does not exist in the candidate set. For example, for the Persian word “متعلّقات/mota‘alleqAt/possession”, three equivalent English words are written in the Aryanpour dictionary: Appurtenance, Paraphernalia, Belongings. In our wordnet, the correct synsets for متعلّقات/possession are determined as follows: {13244109} (noun.possession), property#1, belongings#1, holding#2 – (something owned; any tangible or intangible possession that is owned by someone; “that hat is my property”; “he is a man of property”). However according to FarsNet, the correct synset is {00032613} (noun.Tops) possession#2 – (anything owned or possessed). In this evaluation, the link متعلّقات/possession to synset Noun-02671421 is considered to be incorrect and is penalized.

- The Persian word is not lemmatized correctly; so the English translations and consequently candidate set does not contain the correct synset. For example, the Persian word “باراک/bArAk/Barak” is a proper noun, while the stemmer recognizes بار /bAr/load as its stem, which means “load”.

To resolve the above problems, a set of manually judged links were used in the second experiment. Figure 5 represents the precision and recall of the resulting wordnet for different values of t_{remove} according to manual judges. Parameter t_{remove} demonstrates a threshold, so those links with a score lower than it should be deleted from the final wordnet. High values for t_{remove} result in a wordnet with high precision but low recall. On the other hand, low values for t_{remove} cause a low precision but high recall wordnet. Thus there is a trade-off between precision and recall. For $t_{remove} = 0.1$, the precision of the PMI-based wordnet is 86%, while precision of the wordnet created by the proposed method is 90% according to manual judges. If $t_{remove} = 0$, which means that all links are contained in the final wordnet, the precision is 74%. Therefore, the initial wordnet seen without executing the EM algorithm has 74% precision. Figures 4d and 5c show another quality measure for both wordnets, which is F-measure. Definition of the F-measure and a complete analysis about it is presented in Section 4.3.

4.2.3 Size and Polysemy Rate of the Wordnet

One of the important aspects of wordnets is their size. Large wordnets may have tens of thousands of sysnsets (Miller, 1995; Patanakul & Charnyote, 2005; Black et al., 2006; Piasecki et al., 2011). On the other hand, wordnets with more polysemic words are more useful in NLP and IR tasks. Polysemic words are those words that have more than one sense in the wordnet. Finding the correct sense of polysemic words is of great significance to automatic wordnet construction.



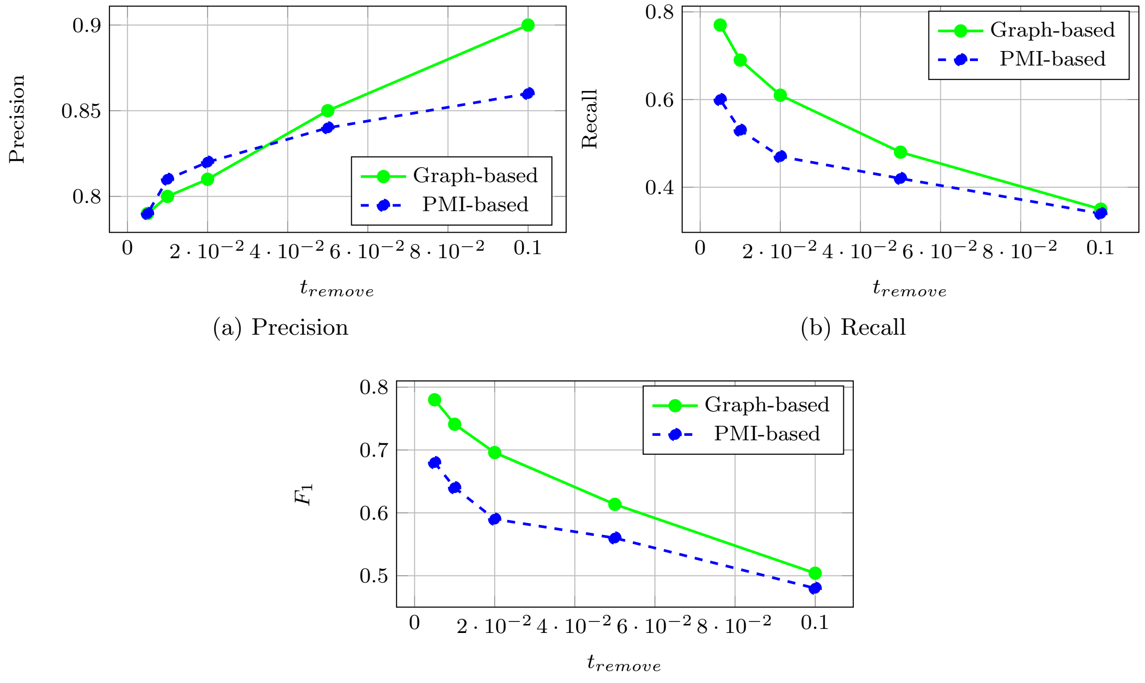
(a) Precision

(b) Recall

(c) Accuracy

(d) F-measure

Figure 4: Comparison between the wordnets according to the FarsNet.



(a) Precision

(b) Recall

(c) F-measure

Figure 5: Comparison between the wordnets according to the manual judges.

Table 4: Comparison between size of the wordnets

	PMI-based wordnet			Graph-based wordnet		
Threshold	unique words	word-synset	polysemy	unique words	word-synset	polysemy
0.1	11,880	27,358	0.63	11,899	29,944	0.73
0.05	11,969	36,922	0.71	11,972	43,690	0.79
0.02	11,974	49,070	0.76	11,972	61,823	0.80
0.01	11,974	58,874	0.78	11,972	74,619	0.80
0.005	11,974	71,761	0.80	11,972	86,879	0.83
0	11,974	141,103	0.85	11,972	141,103	0.85

In this section, the size of the resulting wordnet and polysemy rate for two wordnets, PMI-based and graph-based wordnets, are reported. Table 4 presents the number of unique words, the number of Persian word-WordNet synset links and the proportion of the polysemic words based on different values for t_{remove}. As t_{remove} decreases from 0.1 to 0.01, more unique words will be contained in wordnets, the number of word-synset links increases, and also the proportion of polysemic words to unique words in the wordnet increases. As can be seen, the wordnet created as a result of the graph-based method surpasses the PMI-based wordnet.

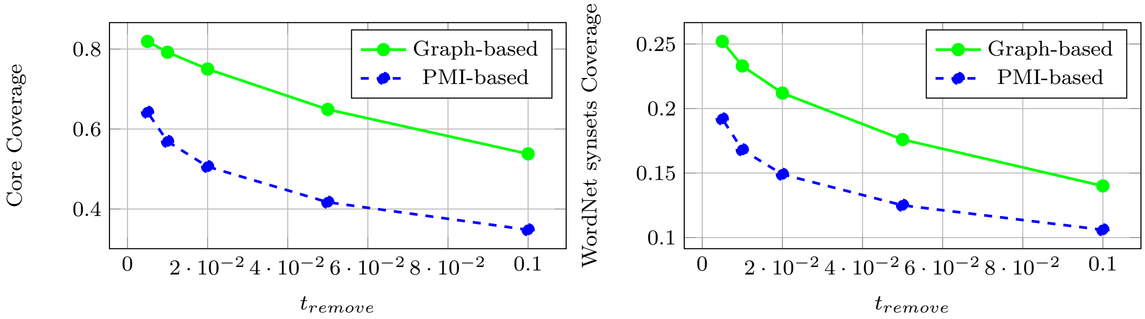
If all links were included in the wordnet, then polysemic words are 85% of the unique words. However, in our wordnet, by removing those links with a probability of less than 0.1, 73% of words are polysemic, which is 10% better than the PMI-base wordnet. For $t_{remove} = 0.1$, both wordnets have about 12,000 unique words. Since both methods were executed on the same corpus, there is no significant difference in their sizes.

4.2.4 Coverage of the Wordnet

To evaluate the coverage of the resulting wordnet, we are interested in observing the coverage over WordNet synsets and also the coverage over language words. In this section, three experiments were performed: 1) core concepts coverage, 2) WordNet synset coverage, and 3) corpus coverage.

In the first experiment, the coverage of our wordnet over core synsets is evaluated. Boyd-Graber et al. (2006) published a list of about 5,000 word-senses in WordNet 3.0, which contains the 5,000 most frequently used word-senses (Core WordNet, 2015). Coverage of a wordnet over this list can be regarded as covering the most common concepts of the language. So this core wordnet was used to measure the percentage of synsets from this list covered by PMI-based and graph-based wordnets. Figure 6a represents the core coverage for different values of t_{remove}. Selecting all links, ($t_{remove} = 0$), causes coverage of 88% of the core wordnet, while choosing links that are more probable than 0.1, leads to coverage of 53% and 34% of the core wordnet for graph-based and PMI-based wordnets, respectively.

In the second experiment, the coverage of wordnets over all WordNet synsets is studied. Since the resulting wordnet is a multi-lingual wordnet, its coverage of it over WordNet synsets is a measure of its quality. Figure 6b represents the coverage of PMI-based and graph-based wordnets over WordNet 3.0 synsets for different values of t_{remove}. This figure shows that the graph-based wordnet covers more WordNet synsets than PMI-based wordnet for all values of t_{remove}. For example, by selecting links with a probability higher than 0.1,



(a) Core Coverage

(b) WordNet Coverage

Figure 6: Coverage of the wordnets over the core synsets and all synsets of the WordNet.

Table 5: Comparison between coverage of the wordnets.

	Coverage over the Bijankhan (unique words)
FarsNet	3,050
PMI-based wordnet	11,523
graph-based wordnet	11,543

the graph-based wordnet covers 14% of WordNet synsets; while the PMI-based wordnet covers 10% of WordNet synsets.

In the third experiment, the coverage of wordnets over the Bijankhan corpus is evaluated. Bijankhan is a large corpus and the proposed method was trained over 13% of it. The rest of this corpus was used for measuring the word coverage of wordnets. Table 5 demonstrates the number of unique words of the corpus, covered by PMI-based and graph-based wordnets, when $t_{remove} = 0.1$. The same evaluation was also performed on FarsNet as the baseline and is also presented in the Table 5. Although the training and testing corpus are separate, there is a significance difference between FarsNet and EM-based wordnets' coverage.

4.3 Parameter Selection

In the proposed method for wordnet construction after convergence of the EM algorithm, a set of links between words of the target language and synsets of the source language is obtained. The links that scored lower than the threshold t_{remove} are removed from the final wordnet. As the previous experiments showed, the value of the t_{remove} affects the quality of the resulting wordnet. The experiments in section 4.2.2 illustrated that changing t_{remove} from 0.005 to 0.1 has a positive effect on the precision but a negative effect on the recall of the resulting wordnet. Indeed, there is a trade-off between the precision and the recall. Here a question may arise; what is the best value for the t_{remove}.

In this section, F-measure is used to investigate the quality of the wordnet, considering both precision and recall. The formula of the F_1 is as follows:

$$F_1 = 2.\frac{precision.recall}{precision + recall}, \tag{8}$$

F_1 is the harmonic mean of the precision and the recall. In order to gain some insight into the optimum value of t_{remove}, the F_1 for the resulting wordnet has been calculated for different values of t_{remove}. As for precision and recall, the F_1 is calculated against both the manual judgement and the FarsNet. Figure 5c shows that F_1 decreases from 77% to 50% when t_{remove} increases from 0.005 to 0.1 for the graph-based wordnet according to the manual judgement. This means that the precision value is more important than that of the recall and the rate at which precision is decreasing is higher than the rate at which recall is increasing. Therefore, to gain a more precise wordnet, we should increase t_{remove}; however, we must accept loosing the recall.

On the other hand, Figure 4d shows that the highest value of the F_1 for the graph-based wordnet is obtained when $t_{remove} = 0.1$ according to FarsNet. This fact means that the recall value has more effect on the F_1 than the precision value. The reason for this difference is due to the low precision values that have been obtained in the evaluation according to FarsNet, as reported in Section 4.2.2. FarsNet lacks most of the correct mappings between the Persian words and the WordNet synsets. Indeed in wordnet construction, the precision of the final wordnet is more important than its recall.

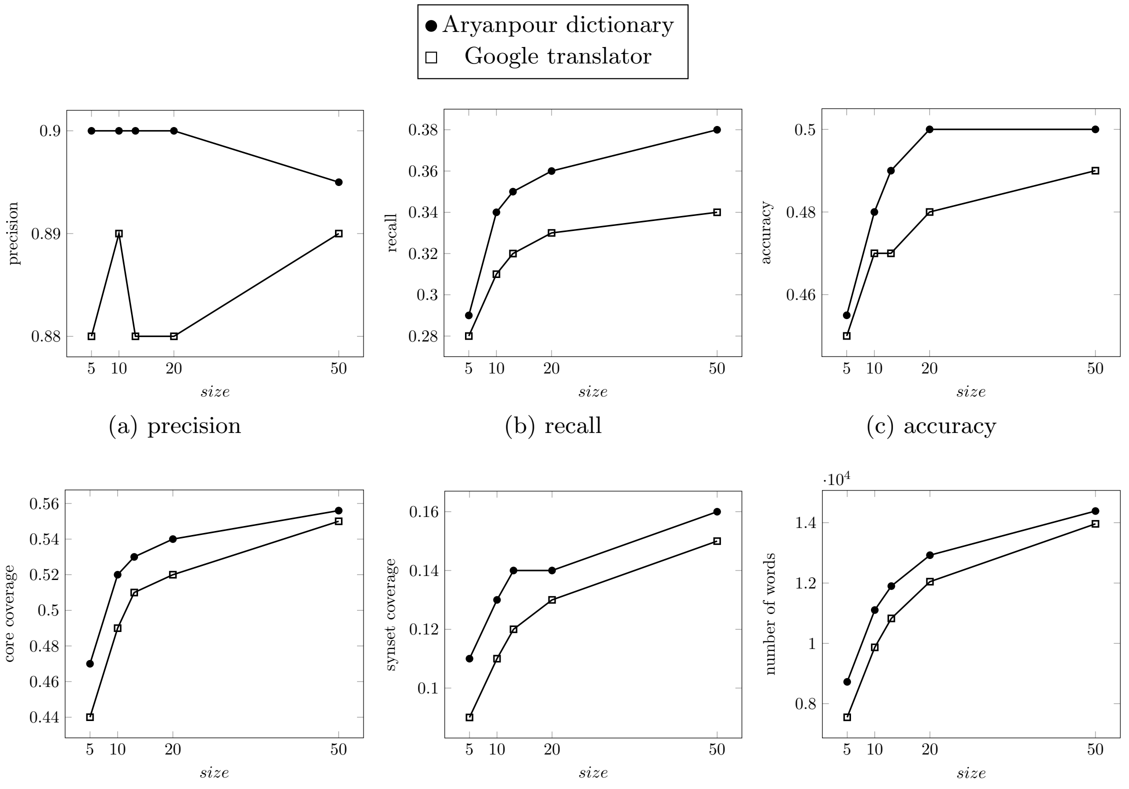
Finally, choosing the threshold t_{remove} has important effect on the quality of the resulting wordnet. However, this matter depends on the application. In most applications, having a more precise wordnet is preferential to having a large but not accurate enough one. In these cases, greater values for t_{remove} is preferential. Although, in applications that high recall is needed, one should choose low values for t_{remove}.

4.4 The Effect of Corpus Size and Dictionary

In this section, the effect of the required resources on the final wordnet is looked at. The proposed method needs a bi-lingual dictionary and a mono-lingual corpus. In the previous experiments, the Aryanpour dictionary and the Bijankhan corpus were used. Since the Bijankhan is a large corpus only 13% of it was used in the previous experiments. To investigate the effect of corpus size on the quality of the resulting wordnet, the proposed method has been executed using four other sizes of the Bijankhan: 5%, 10%, 20% and 50%. Additionally, to examine the effect of the dictionary on the quality of the final wordnet, the Google translator[4] is used in another experiment instead of the Aryanpour; the resulting wordnet is then compared against the wordnet created using Aryanpour on the same size of the Bijankhan. The link removal threshold t_{remove} for all experiments in this section is 0.1. The resulting wordnets have been evaluated for precision, recall, accuracy, coverage over the WordNet core synsets, coverage over all synsets of the WordNet, and the number of the Persian words.

As it is shown in Figure 7, when the size of the corpus increases from 5% to 50% of the Bijankhan corpus using the same dictionary, all measures increase except for precision, which either does not change or changes only slightly. This result is not beyond expectation. Indeed, the precision of the resulting wordnet depends on the precision of the WSD procedure and so does not depend on the size of the corpus. However, new possible senses of the words are discovered by increasing the size of the corpus and therefore the recall, accuracy, coverage and the size of the wordnet increase with growth of the corpus size. As

[4] http://translate.google.com/



(a) precision

(b) recall

(c) accuracy

(d) core coverage

(e) WordNet synset coverage

(f) number of words

Figure 7: Evaluation of the resulting wordnet trained on different sizes of Bijankhan.

Figure 7f demonstrates, to have a wordnet with at least 10,000 words, the corpus size should be at least 10% of the Bijankhan corpus. Figure 7 also illustrates that the wordnet trained on the Aryanpour dictionary excels the wordnet derived from the Google translator. This experiment demonstrates that the dictionary heavily affects the final wordnet even more than the corpus size. As a result, having a small corpus but a large dictionary results in a more precise wordnet than having a large corpus but a small dictionary.

In the last experiment, the proposed method has been executed using the full Bijankhan corpus and the Aryanpour dictionary. The precision, recall and accuracy of the resulting wordnet are 90%, 41% and 52%, respectively. Comparing to the wordnet, which was created using 13% of the Bijankhan and the same dictionary, recall and accuracy increased 6% and 3%, accordingly; while the precision does not change. This wordnet has 15,406 Persian word and covers 61% of the core synsets of the WordNet. Considering all synsets of the WordNet, it covers 20% of them.

5. Conclusion

In this paper, an EM algorithm was employed in order to develop a wordnet for low-resourced languages. We successfully applied unsupervised cross lingual WSD in the expectation step of the algorithm. The proposed method does not use any features specific to the target language, and so it can be used for other languages to generate wordnets. Resources needed for this proposed algorithm include a bi-lingual dictionary and a mono-lingual corpus. The proposed method belongs to the expansion approach and so creates a multi-lingual wordnet in which for each word in the target language, the equivalent synset in WordNet is known.

The proposed method was applied on the Persian language and the quality of the resulting wordnet was examined through several experiments. Its precision was 18% according to FarsNet and 90% according to the manual judgement. The reason for this difference is that the WordNet synsets are too fine-grained in comparison to the FarsNet synsets, and so most of the synsets in FarsNet should be mapped onto more than one synset in WordNet; however FarsNet provides one or at most two WordNet synsets for most of FrasNet synsets. This problem means that most of the correct links in the resulting wordnet are considered to be incorrect and thus the reported precision becomes low. Also, the resulting wordnet contains about 12,000 words of the Persian language from only using 13% of the Bijankhan corpus, which is more than several wordnets in other languages. Additionally, 53% of core synsets and 14% of all synsets of WordNet are covered. Analysis of the effects of corpus size and dictionary size of the resulting wordnet showed that the dictionary size can affect the precision of the wordnet more than the corpus size and therefore it is important to use large-enough dictionaries.

Acknowledgements

This research was in part supported from Institute for Research in Fundamental Sciences (No. CS1395-4-19).

References

Apidianaki, M., & Sagot, B. (2014). Data-driven synset induction and disambiguation for wordnet development. *Language Resources and Evaluation*, *48*(4), 655–677.

Atserias, J., Climent, S., Farreres, X., Rigau, G., & Rodríguez, H. (2000). Combining multiple methods for the automatic construction of multilingual wordnets. *Amsterdam studies in the theory and history of linguistic science series 4*, 327–340.

Barbu, E., & Barbu Mititelu, V. (2005). A case study in automatic building of wordnets. In *Proceedings of OntoLex 2005- Ontologies and Rexical Resources*, pp. 85–90, Jeju Island, Korea. Asian Federation of Natural Language Processing.

Basile, P., Caputo, A., & Semeraro, G. (2014). An enhanced lesk word sense disambiguation algorithm through a distributional semantic model. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pp. 1591–1600, Dublin, Ireland. International Committee on Computational Linguistics.

Black, W., Elkateb, S., & Vossen, P. (2006). Introducing the Arabic wordnet project. In *Proceedings of the Third International WordNet Conference (GWC-06)*, pp. 295–299, South Jeju Island, Korea. Global WordNet Association.

Bond, F., & Foster, R. (2013). Linking and extending an open multilingual wordnet. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pp. 1352–1362, Sofia, Bulgaria. Association for Computational Linguistics.

Bond, F., Isahara, H., Kanzaki, K., & Uchimoto, K. (2008). Boot-strapping a wordnet using multiple existing wordnets. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, pp. 1619–1624, Marrakech, Morocco. European Language Resources Association (ELRA).

Boudabous, M. M., Chaâben Kammoun, N., Khedher, N., Belguith, L. H., & Sadat, F. (2013). Arabic wordnet semantic relations enrichment through morpho-lexical patterns. In *Proceeding of 1st International Conference on Communications, Signal Processing, and their Applications (ICCSPA)*, pp. 1–6, American University of Sharjah, United Arab Emirates. IEEE.

Boyd-Graber, J., Fellbaum, C., Osherson, D., & Schapire, R. (2006). Adding dense, weighted connections to WordNet. In *Proceedings of the third International WordNet Conference (GWC-06)*, pp. 29–35, South Jeju Island, Korea. Global WordNet Association.

Buitelaar, P., & Cimiano, P. (2014). *Towards the Multilingual Semantic Web.* Springer Berlin Heidelberg.

Cohen, J. (1960). A coefficient of agreement for nominal scales". *Educational and Psychological Measurement*, *20*(1), 37–46.

Core WordNet (2015) `http://wordnetcode.princeton.edu/standoff-files/core-wordnet.txt`.

Diab, M. (2004). The feasibility of bootstrapping an Arabic wordnet leveraging parallel corpora and an English WordNet. In *Proceedings of the Arabic Language Technologies and Resources*, Cairo, NEMLAR.

Dini, L., Peters, W., Liebwald, D., Schweighofer, E., Mommers, L., & Voermans, W. (2005). Cross-lingual legal information retrieval using a WordNet architecture. In *Proceedings of the 10th international conference on Artificial intelligence and law (ACAIL)*, pp. 163–167, Bologna, Italy. ACM.

Elkateb, S., Black, W., Rodríguez, H., Alkhalifa, M., Vossen, P., Pease, A., & Fellbaum, C. (2006). Building a wordnet for Arabic. In *Proceedings of The 5th international conference on Language Resources and Evaluation (LREC 2006)*, Genoa, Italy. European Language Resources Association (ELRA).

Erjavec, T., & Fišer, D. (2006). Building Slovene wordnet. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, Genoa, Italy. European Language Resources Association (ELRA).

Fellbaum, C., & Vossen, P. (2012). Challenges for a multilingual wordnet. *Language Resources and Evaluation*, *46*(2), 313–326.

Fišer, D. (2009). Human language technology. In *Leveraging Parallel Corpora and Existing Wordnets for Automatic Construction of the Slovene Wordnet*, pp. 359–368. Springer Berlin Heidelberg.

Gunawan, G., & Saputra, A. (2010). Building synsets for Indonesian wordnet with monolingual lexical resources. In *Proceedings of International Conference on Asian Language Processing (IALP)*, pp. 297–300, Harbin, China. IEEE.

Hasanuzzaman, M., Caen, F., Dias, G., Ferrari, S., & Mathet, Y. (2014). Propagation strategies for building temporal ontologies. In *Proceedings of 14rd conference on European Chapter of the Association for Computational Linguistics*, pp. 6–11, Guthenburg, Sweden. Association for Computational Linguistics.

Kaji, H., & Watanabe, M. (2006). Automatic construction of Japanese wordnet. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, Genoa, Italy. European Language Resources Association (ELRA).

Kazakov, D., & Shahid, A. R. (2009). Unsupervised construction of a multilingual wordnet from parallel corpora. In *Proceedings of the Workshop on Natural Language Processing Methods and Corpora in Translation, Lexicography, and Language Learning*, pp. 9–12, Borovets, Bulgaria. Association for Computational Linguistics.

Lam, K. N., Al Tarouti, F., & Kalita, J. (2014). Automatically constructing wordnet synsets. In *52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*, pp. 106–111, Baltimore, USA. Association for Computational Linguistics.

Landes, S., Leacock, C., & Tengi, R. I. (1998). Building semantic concordances. *WordNet: an electronic lexical database*, *199*(216), 199–216.

Mallery, J. C. (1988). *Thinking about foreign policy: Finding an appropriate role for artificially intelligent computers.* Ph.D. thesis, MIT Political Science Department.

Miller, G. A. (1995). WordNet: a lexical database for english. *Communications of the ACM*, *38*(11), 39–41.

Montazery, M., & Faili, H. (2010). Automatic Persian wordnet construction. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pp. 846–850, Beijing, China. Association for Computational Linguistics.

Montazery, M., & Faili, H. (2011). Unsupervised learning for Persian wordnet construction. In *Proceedings of Recent Advances in Natural Language Processing (RANLP)*, pp. 302–308, Hissar, Bulgaria. Association for Computational Linguistics.

Navigli, R. (2009). Word sense disambiguation: A survey. *ACM Computing Surveys (CSUR)*, *41*(2), 10.

Navigli, R., & Lapata, M. (2010). An experimental study of graph connectivity for unsupervised word sense disambiguation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *32*(4), 678–692.

Navigli, R., & Ponzetto, S. P. (2010). BabelNet: Building a very large multilingual semantic network. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pp. 216–225, Uppsala, Sweden. Association for Computational Linguistics.

Navigli, R., & Ponzetto, S. P. (2012a). BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, *193*, 217–250.

Navigli, R., & Ponzetto, S. P. (2012b). Multilingual WSD with just a few lines of code: the BabelNet API. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL 2012)*, pp. 67–72, Jeju, Republic of Korea. Association for Computational Linguistics.

Oliver, A., & Climent, S. (2012). Parallel corpora for wordnet construction: machine translation vs. automatic sense tagging. In *Proceedings of the 13th International Conference on Intelligent Text Processing and Computational Linguistics*, pp. 110–121, New Delhi, India. Springer.

Oroumchian, F., Tasharofi, S., Amiri, H., Hojjat, H., & Raja, F. (2006). Creating a feasible corpus for Persian POS tagging. Tech. rep. TR3/06, University of Wollongong, Dubai.

Otegi, A., Arregi, X., Ansa, O., & Agirre, E. (2015). Using knowledge-based relatedness for information retrieval. *Knowledge and Information Systems*, *44*(3), 689–718.

Patanakul, S., & Charnyote, P. (2005). Construction of Thai wordnet lexical database from machine readable dictionary. In *Conference Proceedings: the tenth Machine Translation Summit*, pp. 87–92, Phuket, Thailand. Language Technology World.

Piasecki, M., Kurc, R., & Broda, B. (2011). Heterogeneous knowledge sources in graph-based expansion of the Polish wordnet. In *Intelligent Information and Database Systems*, Vol. 6591, pp. 307–316. Springer.

Prabhu, V., Desai, S., Redkar, H., Prabhugaonkar, N., Nagvenkar, A., & Karmali, R. (2012). An efficient database design for IndoWordNet development using hybrid approach. In *Proceedings of the 3rd Workshop on South and Southeast Asian Natural Language Processing (SANLP)*, pp. 229–236, Mumbai, India. International Committee on Computational Linguistics.

Rodrquez, H., Farwell, D., Ferreres, J., Bertran, M., Alkhalifa, M., & Mart, M. A. (2008). Arabic wordnet: Semi-automatic extensions using Bayesian inference. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Saveski, M., & Trajkovski, I. (2010). Automatic construction of wordnets by using machine translation and language modeling. In *Proceedings of the 13th International Multiconference*, pp. 78–83, Ljubljana, Slovenia. Information Society.

Semantically Tagged glosses (2016) `http://wordnet.princeton.edu/glosstag.shtml`.

Shamsfard, M. (2008). Towards semi automatic construction of a lexical ontology for Persian. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Shamsfard, M., Hesabi, A., Fadaei, H., Mansoory, N., Famian, A., Bagherbeigi, S., Fekri, E., Monshizadeh, M., & Assi, S. M. (2010a). Semi automatic development of FarsNet; the Persian wordnet. In *Proceedings of 5th Global WordNet Conference*, Mumbai, India. Global WordNet Association.

Shamsfard, M., Jafari, H. S., & Ilbeygi, M. (2010b). STeP-1: A set of fundamental tools for Persian text processing. In *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC 2010)*, Valletta, Malta. European Language Resources Association (ELRA).

Tufis, D., Cristea, D., & Stamou, S. (2004). BalkaNet: Aims, methods, results and perspectives. a general overview. *Romanian Journal of Information Science and Technology*, *7*(1-2), 9–43.

Vossen, P. (1998). Introduction to EuroWordNet. In *EuroWordNet: A multilingual database with lexical semantic networks*, pp. 1–17. Springer.