

Automatic Translation between English and Persian Texts

Chakaveh Saedi^{1,2}, Mehrnoush Shamsfard¹, Yasaman Motazed^{1,2}

¹NLP Research Lab., Electrical & Computer Engng Dept., Shahid Beheshti University, Tehran, Iran

²Azad University, Science and Research Branch, Tehran, Iran.

{y_motazed, m-shams, ch_saedi}@sbu.ac.ir

Abstract

PEnTrans is an automatic bidirectional English/Persian text translator. It contains two main modules, PEnT1,2. PEnT1 translates English sentences into Persian and PEnT2 performs translation from Persian to English. WSD which is an important part in translation is done in both systems by employing a combination of extended dictionary and corpus based approaches in PEnT1 and employing a combination of rule based, knowledge based and corpus based approaches in PEnT2. In this paper, introducing PEnTrans and its components, we propose a new WSD method by presenting a hybrid measure to score different senses of a word according to its condition in a sentence and other words in the sentence.

1 Introduction

Machine Translation is one of the interesting fields in NLP having a vast usage, from translation of a simple web page to domain-specific texts. Regardless of all the quarrels for the capability of computers for translation in the first years of its overture, now it has a good credit and numerous reliable systems (bilingual and multilingual) are implemented.

There are different developed methods and approaches in machine translation and beside them there are some parameters upon which we can decide on the suitable approach to translate between two languages. Some of these parameters are (1) availability of linguistic resources such as semantic lexicons, thesauri, parallel or tagged corpora and ontologies (2) structural and lexical similarity of two languages and (3) the desired depth and precision of translation. For example if one wants to perform corpus based translation he/she should have appropriate corpora otherwise exploiting data driven methods is not a good decision. When we need a deep level of understanding, knowledge based semantic methods perform better than statistical methods and so on. In other words knowledge based approaches are

mostly employed where either we need deep processing of texts or there is no reliable corpus available.

Although there are many translation systems developed to translate from/to English, there have been very restricted efforts in developing translators from/to Persian. Persian is an Indo-European language which is the official language of Iran, Afghanistan and Tajikistan. The modern Persian as written in Iran is a right to left script which looks like Arabic script but it has its own alphabet and grammatical rules. The shortcomings of efficient, reliable linguistic resources and fundamental text processing modules for Persian language, make it hard to be processed by computers. In recent years there have been two branches of efforts to eliminate these shortcomings. Some researchers are working to provide linguistic resources and fundamental processing units and some others work on developing shallower algorithms with less need to these essentials.

2 Persian translation: challenges and problems

In Persian translation, there are some general problems and difficulties inherited from general text processing field such as natural language ambiguities, anaphora resolution, processing idioms and slangs, etc. Some other difficulties come into being or become more severe for processing Persian language. Some of them are as follows:

- *Lexical problems*: Usually none of the short vowels are written in a Persian sentence. So encountering homographs and homonyms is a popular ambiguity in Persian. Furthermore as in Persian a short vowel (called Ezafe) is used to correlate two nouns and so in most cases is not written in text (except for special cases), determining the relation between nouns and their modifiers (nouns or adjectives) is a challenge in Persian sentences. On the other hand Persian is a derivative and generative language in which many

new words may be built by concatenating words and affixes. This causes many challenges especially for recognizing compound verbs.

- *Segmentation Ambiguity*: There are one to four written forms for each character according to its location in a word. Also there are various scripts for writing Persian texts, differing in the style of writing words, using or elimination of spaces within or between words, using various forms of characters and so on. These all make Persian texts hard to be processed by computer. So tokenization which is one of the early steps of text preprocessing is a complex and challenging part in Persian language (Kiani, 2008).
- *Structural Ambiguity*: Although Persian has a canonical SOV order, there are lots of frequent exceptions in word order, caused by processes such as scrambling, verb preposing, postponing, dislocation, clefting and pseudoclefting and topicalization movement (Mahootian, 1997) and result in high structural ambiguity.

The other problems are related to the differences between Persian and the other side of translation (English here) such as:

- *Morphological differences*: In Persian even words which are uncountable may appear in plural form.
- *Verbs*: In Persian construction of present stems from infinitives are mostly irregular. Many compound verbs can be derived from nouns and adjectives and in many cases the parts of these verbs are separated by some words and have long distance dependencies.
- *Structural and Lexical Gaps*: Subject may be omitted in Persian sentences while they should be present in English (except for imperatives). English uses auxiliary verbs for negation and interrogation, while there is no auxiliary verb in Persian. Usually there is no definite article in Persian while most of the nouns appear with one in English. Unlike English there is no female/male distinction for Persian pronouns while the gender must be considered for applying “she” and “he” in English.

This paper introduces an MT system called PEnTrans. PEnTrans aims at building a bidirectional bilingual machine translation system capable to translate between Persian and English texts (in both directions). It translates stand alone sentences regardless of their discourse and surrounding sentences. PEnTrans consists of two parts; PEnT1 and PEnT2.

PEnT1 translates simple English sentences into Persian, exploiting a combination of rule based and semantic approaches. The main feature of PEnT1

which distinguishes it from other related works is exploiting a novel semantic Word Sense Disambiguation (WSD) algorithm which uses WordNet, eXtendedWordNet and verbs part of FarsNet to find the appropriate sense of input words.

PEnT2 is the Persian to English side of PEnTrans. It translates simple Persian sentences to English. By simple sentences we mean either sentences including just one verb or co-ordinate non-crossover sentences. There is no limitation in the number of words included in the input sentence.

PEnT2 uses a hybrid approach; a combination of rule based, knowledge based and corpus based approaches and employs the grammatical role of words in the sentence as the main clue to perform the translation process according to the available knowledge.

3 Related works

Different works have been done on knowledge based approaches such as how to represent the knowledge (Hahn, 2003), what are the different types of it (Agirre and Martinez, 2001) and combination of knowledge based approach with other approaches (Montoyo, et al. 2005). There have been some research activities to develop English/Persian translators. Using Tree Adjoining Grammars (Feili and Ghassem-Sani 2004) and application of Lexicalized Grammars in English-Persian Translation (Feili and Ghassem-Sani 2004) are some of these research works. There are also some commercial products performing English to Persian translation. The commercial translators, to name: Pars and Aria, have not meet the minimum machine translation metrics. They have too many problems especially facing complex phrases/sentences and also word sense disambiguation. There are also a few works on translating from Persian to English (Shiraz, Language Weaver), most of which do not have high performance. The shortage of Persian text processing tools and resources such as reliable Persian parsers, lexical ontologies, complete electronic Persian thesauri, parallel corpora and even complete computational bilingual dictionaries have been some of the problems encountered in developing Persian/English translators.

4 Introduction to PEnTrans

PEnTrans is a bidirectional automatic machine translation system which translates between English and Persian. The architecture of this system is shown in Figure1. There are three main components in this

system. **Input analysis** which tries to extract every single word of the sentence and the important information about them, **Lexical transfer** which aims to find the best translation for each word and **Structural transfer** which puts the extracted

translations in target language structure. In the next sections, we will explain PEnT1,2 structures and more descriptions of these 3 components are given.

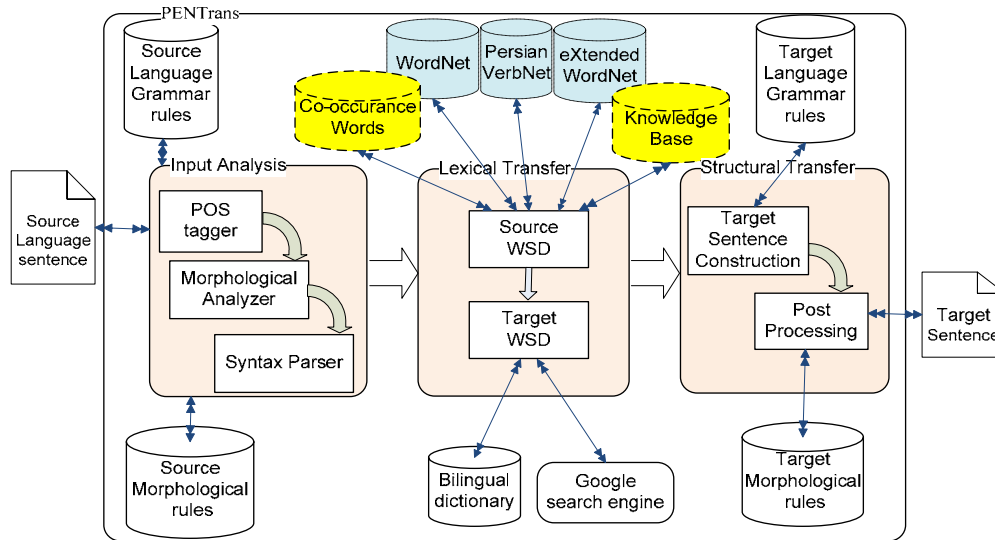


Figure 1. PEnTrans architecture (WordNet, Persian VerbNet and eXtended WordNet are employed in PEnT1 while the dashed resources are PEnT2's and the rest are shared between both modules)

5 English to Persian translation in PEnT1

“PEnT1” is the English to Persian module of PEnTrans. It is capable of translating simple sentences, Including: imperative, negative and interrogative in both active and passive speech. By “Simple sentence” we refer to any type of sentence that contains one main verb or at most two verbs with the subject. For example “I want to stay.” is an admissible one while “She wants me to stay.” is not. All the twelve tenses are correctly recognized and translated into their equivalences by PEnT1.

It can be seen that other developed systems or proposed approaches pay little attention to WSD or implement it just as a prototype for a specific domain. On the whole, they are highly error-prone. As the few works on WSD for English to Persian translation, we can mention ‘Using a Decision Tree Approach for Ambiguity Resolution’ (Feili and Ghassem-Sani 2005) and ‘using target language corpus for WSD task’ (Mosavi and Delavar 2005). The first one defines some decision trees to find the most proper Persian equivalent using POS tag of elementary tree anchor and the specific feature of anchor and none anchor nodes. The second one takes a corpus-based

approach considering the one or two neighbor word(s) for each ambiguous word. The results of this paper seem to be admissible. But as it uses a domain-specific corpus, it can be concluded that if domain extends, then the precision might be decreased and yet, there is no domain specific corpora with Persian POS tags available.

There are many works on WSD in the English side. One of the most famous is the Lesk algorithm (Lesk 1986) for which we have proposed an extension. There are some other variations of Lesk algorithm using WordNet as their lexical knowledge base, namely (Banerjee and Pedersen 2002) and (G.Ramakrishnan, B et al. 2004). Our extension is to consider taxonomic relations in the WordNet and parsed glosses of eXtendWordNet beside other parameters of Lesk. We also introduce a new scoring method to calculate the similarity measure and find the most appropriate sense for ambiguous words.

5.1 PEnT1 architecture

As discussed in section 2, PEnT1 has three main components. Each of these components with their specific resources and responsibilities is discussed in the following subsections in detail.

5.1.1 Input Analysis

PEnt1 uses Stanford syntax parser (Christopher D. Manning 2003; Christopher D. Manning 2003) to perform tokenization, syntax parsing and some part of morphological analysis, in the first step. PEnt1 uses the output containing tokens (words) and their POS tags, the parse tree and the extracted syntactic relations (Marneffe et al. 2006) between different constituents of the sentence.

Regarding English and Persian syntactic differences, such as Subject-Verb-Object or noun and adjective orders, syntax parser's output is not appropriate input for the generation phase. Some tasks needed to be done prior to transfer are as follows:

- The main verb of the sentence and its specifications such as tense, voice (active/passive), transitivity and person are determined.
- Stop word recognition and handling for the procedure of WSD and Lexical Transfer.

WSD refers to selection of the most appropriate sense (meaning) of a word in the context and is considered as one of the most important tasks in machine translation. In this field, ambiguity resolution or WSD should be performed in two dimensions: (1) to find the appropriate sense of a word in the source sentence and (2) to find the best translation of this sense in the target sentence.

5.1.2 Source Word Sense Disambiguation

WSD in the source language (English) is achieved by implementing some extensions to Lesk algorithm (Lesk 1986) using WordNet (Fellbaum 1998; Banerjee and Pedersen 2002) and eXtended wordNet (Harabagiu, Miller et al. 1999). We proposed four WSD methods, introduced in section a to d.

a) WSD-S: Extended Lesk Algorithm: Based on Lesk algorithm, for any given ambiguous word, the glosses of all of its senses is compared to the glosses of every other words in the sentence. A word is assigned that sense whose gloss shares the largest number of words in common with the glosses of the other words. This algorithm consider all synsets and glosses (including provided exmples) of each word. In our extended algorithm we assign different weights (scores) to words occurring in the synset or occurring in the gloss. Therefore, instead of counting the number of shared words between senses, we calculate the sum of the scores of common words. Finally set of senses with maximum score is introduced as the WSD result.

b) WSD-E1: The results of applying WSD-S algorithm on test sentences show that there are cases in which common words refer to different senses (homographs and polysemy or even different POS). To solve this issue, part-of-speech and sense tags of each word is considered for scoring. These tags are retrieved from eXtended WordNet. Therefore two words get the full score of similarity just in case that their stem, POS and sense tag are the same; otherwise a penalty would be applied. The side effect of using eXtended WordNet is reduction in the number of comparing words for each sense due to POS tagging and exclusion of example words. To avoid data sparseness we took another approach named WSD-E2.

c) WSD-E2: We borrowed the idea of moving in hierarchy from Banerjee and Pedersen (2002) in hope of providing more related and tagged words for each sense. Although we stop at the second level hypernym to prevent slowness of algorithm.

d) WSD-PB: The above algorithms show good performance for sentences having a few ambiguous words and even those with 4-7 ambiguous words, each having at most 10 senses. As the search space grows, the algorithm slows down. To reduce the space complexity, a semantic approach is taken. It compares just the set of words which have grammatical relations with each other according to the output of Stanford parser. Consider sentence 1 as an example. There are three ambiguous words: bake, savory and pancake. Two relations are defined as follows: (1) amod (pancake, savory) and (2) dobj (back, pancake). Therefore, if there exists a common word between any gloss/synset of word "savory" and "bake", it is not considered in scoring.

- (1) We will bake savory pancakes.

5.1.3 Target Word Sense Disambiguation

For doing WSD in the target language, an English-Persian dictionary presenting exact translation of each sense is needed. In other words we need the translations of English synsets appeared in WordNet. As there is neither a Persian WordNet correspondence to the English one nor a database of Persian translations of WordNet synsets available, we created a portion of Persian WordNet prototype and used it to test our algorithms. We also used verbs module of FarsNet project (Rouhizadeh 2008) to obtain present and past tenses of Persian verbs.

As indicated before the Persian equivalent of each word is extracted from a prototyped dictionary which

is a mapping of WordNet. Whenever the equivalent synset contains more than one word, some queries are generated, containing the ambiguous word and its neighbors, and sent to Google search engine to find the most natural combination of words.

5.1.4 Structural transfer

PEnT1 Generation Phase refers to Persian parse tree construction followed by a post processing procedure which is described in the following subsections.

Each parse tree contains some phrases which are independent and can be translated autonomously in the sentence. Concerning this trait, Persian parse tree is constructed in a bottom-up approach. This means that in the first step, the word order in each phrase is determined using Persian grammatical rules, then the relative location of phrases is determined and finally the Persian parse tree is constructed based on Persian syntactic rules. In Figure 2 some of PEnT1's customized rules are shown which are defined based on parser's results. Figure 3 shows some English noun phrases and their equivalent Persian parse trees. It must be considered that Persian trees should be read from right to left, and English trees from left to right.

Rule1: If there is more than one noun in a single noun phrase, the last one should be considered as the head. (e.g. in "book fair" the head is "fair") So the other words should be reordered based on the head according to the following rules:
Rule2: All the adjectives should be placed after the head.
Rule3: All the nouns should be placed after the head and its adjectives.
Exception: superlatives and ordinal numbers should be placed before head in Persian

Figure 2. Some of the customized rules based on parser result.

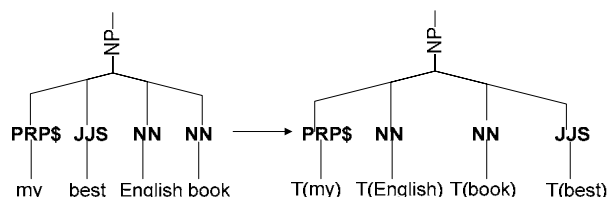


Figure 3. Structural Transfer in PEnT1

The last step in the translation phase is combining WSD output with constructed Persian tree by putting the correct translation of each word in its place.

As indicated previously, all the nouns and verbs are in their single form. To have a sensible sentence we have to add morphological information extracted in the preprocessing phase to the corresponding word(s).

During this post process some Persian inflectional and orthographic rules should be applied.

5.2 Experimental Results of PEnT1

To evaluate our work, a set of test sentences were prepared which cover a range of simple to complicated phrases. Then we have performed three types of tests. In the first type we evaluated our WSD algorithms and compared them with a golden standard created by 5 human evaluators for 170 sentences. Results can be seen in Table 1, where average indicates output accomplishment. In the first type we evaluated our WSD algorithms and compared them with a golden standard created by 5 human evaluators for 170 sentences. Results can be seen in Table 1.

Algorithm	Average
WSD-S	73.65%
WSD-E1	86.65%
WSD-E2	79.9%

Table 1. WSD test results

In the second type of tests, we checked the generated translation according to its grammatical soundness and meaning transfer independently, Table 2 shows the results.

Sentence Type	Meaning transfer	Grammatical Correctness
Indicative	93.3%	93.25%
Imperative	93%	92.8%
Interogative	96.3%	95.25%

Table 2. Comparing to Golden standard test results

In the third type of tests, we compared our system with available commercial systems. Table 3 shows the results and obviously indicates the superiority of PEnT1 over the others in WSD results. (PEnT1 score is the average of scores introduced in Table 2).

Sentence Type	PEnT1	Pars	Aria
Indicative	93.27%	86.2%	81%
Imperative	91.7%	82.5%	83.8%
Interogative	95.7%	84%	77.9%

Table 3.comparison between PEnT1 and two other available translators

6 Persian to English Translation in PEnT2

PEnT2 as a Persian to English machine translation system with its main components (Input analysis, word transfer and structural transfer) are discussed in the following sections.

6.1 Input analysis

The first phase of translation is done in the input analysis component. This component consists of tokenizer, POS tagger, morphological analyzer and a syntax parser. In this part we used some available tools such as (Kiani, 2008) for tokenization and also for inflectional morphological analysis. Then a shallow parser finds the grammatical roles of the constituents and passes the extracted information to the next component.

The output of this step is a sequence of the words' stems tagged by their grammatical roles (subj, obj, ...), inflectional information of the words and some labels like proper nouns or being a human. These labels lead in better translation and we use them in some of our WSD or structural transfer rules later.

6.2 Lexical transfer

Lexical transfer aims to find the corresponding translation for each individual word of the sentence. Ambiguity is the most important problem of this phase and occurs when a Persian word has more than one sense (meaning) and/or more than one English translation. Our proposed WSD algorithm solves both.

6.2.1 Word Sense Disambiguation (WSD)

Word sense disambiguation in MT is mostly done by statistical methods and using co-occurrence words (Wilks and Nirenburg, 2009). Although these methods are usually portable and easy to implement, the main disadvantages of them are being time consuming and highly error prone and also their need to large linguistic resources (Zhou and Han, 2005).

As Persian text processing suffers from the lack of computational linguistic resources such as semantic lexicons, parallel corpora and lexical ontologies, in this system we avoid employing those methods as far as possible. We rather use a knowledge based approach to WSD problem, providing our own essential knowledge base. In our proposed approach, there are two steps to disambiguate words:

1- *Using heuristic rules*: We defined some heuristic rules to find the appropriate sense for a given ambiguous word. These rules are based on two main factors: the word itself and its neighbors and three features of these factors: grammatical role, POS tag and co-occurrence words. So we defined 8 categories of first level WSD heuristic rules corresponding to the word itself, the POS tag of the word, the grammatical role of the word, the location of the word, the collocating words (neighbors), the POS tag of the

neighbor, the grammatical role of the neighbor and the conceptual category of the neighbor (we defined 20 conceptual categories). The words included in the knowledge base have been extracted from Wikipedia sentences, children's Sokhan dictionary (Anvari and Gazarani, 2004) and test sentences. This collection is extracted with a cooperative method.

To resolve the ambiguity, PEnT2 searches the knowledge base to find a rule which matches the ambiguous word's condition. Rules are sorted due to their importance and frequency of detection in out tests so deciding among the results of overlapping rules would be easier. Table 4 shows the applied rules for the word [ساعت, *sa'at*, hour/o'clock/ clock/ watch] :

translation	rule
O' clock	1- Checking If there is a number after it
Hour	2- Checking if there is a number before it
Hour	3- Checking if the word is Adverb of time
Clock/watch	4- Checking if the word is the e subject in the sentence
Clock/watch	5- Checking if the word is the e object in the sentence
Hour	6- "hour" if the other rules condition are not met

Table 4. WSD rules for [ساعت, *sa'at*, hour/o'clock/clock/watch]

As there may be more than one occurrence (with different senses) of the ambiguous word in an input phrase, we use some linguistic knowledge and scoring methods while applying the neighbor itself (with no consideration on the grammatical role and ...). For example in the following sentence there are two occurrences of word (شیر, *shir*) with two different senses (one as Lion and one as milk).

(2) او شیر داغ را در قفس شیر گذاشت

Oo shir-e dagh ra dar gafas-e shir gozasht,

'He put the hot milk in the lion's cage.'

The co-occurring words are the same for both so we have to score them according to some parameters such as distance between the ambiguous word and its co-occurrences or belonging to the same grammatical constituent. For instance if the co-occurred word (B) is an adjective and we have A_1 and A_2 as two occurrences of an ambiguous word.

We have the following rule for scoring A_i :

$$\text{Score} \begin{cases} \text{Max} & \text{if } A_i \text{ and B are in a noun phrase} \\ 0 & \text{if } A_j \text{ and B are in a noun phrase (i!=j)} \end{cases}$$

2- *Searching the Web corpus*: If there is any remained ambiguous word after the last steps, PEnT2 employs the web as a large corpus for WSD. This phase will be performed after structural transfer when the complete English sentence is almost produced. This step is searching the complete English sentences or a part of it in Google in order to compare the number of hits. In each search one of the possible translations of the ambiguous word is applied and at last the translation with the most number of hits will be selected. For instance “high/tall/long” are all translations of the word [بلند, *boland*]. But number of search results for “tall girl” is much more than the results for “high girl” or “long girl”! So the best translation here would be “tall”.

Different scoring rules are applied here too. The full sentence has the most score but as it may not be found on the web, we choose smaller constituents such as noun phrases containing the ambiguous word (with lower score) too.

6.3 Structural transfer

To make a correct sentence we need to arrange the translated words in a new order considering all differences between the two languages.

We chose some English language credible structures such as the following ordering for active voices: [condition + subjP + aux_verb + verb + objP + prepP + adverbs]

The English sentence will be built in a bottom-up manner by creating and putting the corresponding grammatical constituents in their appropriate position.

Besides structural transfer we need some pre/post processing tasks to optimize the output. To do this task, we defined more than 50 rules categorized in 13 main classes (according to 13 main grammatical roles) and 6 supporting classes. The main rules are applied to place the translation of each grammatical role in its appropriate position and the supporting rules are used either for preparing the Persian phrases to be transferred to English or optimizing the produced English sentences.

6.4 Experimental results of PEnT2

To evaluate PEnT2 four different tests were run:

1- WSD tests: To create the knowledge base, we extracted all ambiguous words from a children dictionary (Anvari and Gazarani, 2004) which made 175 ambiguous words with 481 different senses. Then we defined 308 rules (manually) to handle them. Two different test sets were used. In the first set, just 30

random ambiguous words were appeared in all of their senses in 200 complicated test sentences created by 10 reviewers, while in the second test almost all of the studied ambiguous words (with all of their senses) were under test in a large corpus¹. The result is summarized in table5.

test	No. of sentences	No. of ambiguous words occurrences	Correct WSD #	Accuracy(%)
1	200	235	186	79%
2	155700	287	264	91%

Table 5. WSD tests results

2- Understandability test: This test aimed to control both lexical transfer and structural transfer modules. As we had no parallel corpus we selected 170 sentences covering all acceptable structures and ambiguous words and translated them by PEnT2 and gave the outputs to 12 reviewers again to translate them back to Persian. Afterwards the similarity between the original Persian sentences and those which were translated by humans were examined manually. Table 6 shows the result of this test.

Total sentence #	170
Completely similar	84.7%
Similar in meaning but different words	7%
Almost understandable	7%
Non-Admissible translation	1.3%

Table 6. Understandability test Results

3- Comparing with other systems: Persian to English MT is a field in which there have been minimal researches done, like the Shiraz project which is given up in 1999 (Amtrup et al. 2000) and Language Weaver Project which is a commercial system. There are just small demos of these systems available for which we ran PEnT2 too. The result of comparison shows better results in PEnT2 translation according to both syntactic structure of the output and the result of WSD process.

Circular test: In order to examine both systems together, we performed a circular test in which a source language sentence (e.g. English sentence) was translated by PEnTi (e.g. i=1) and the output was retranslated to the source language by the other system (e.g. PEnT2). Then a comparison was done between the original sentence and the ultimate output. Result shows admissible performance.

¹ <http://ece.ut.ac.ir/DBRG/Bijankhan/>

Error analysis: In our tests, 25.5% of errors were for adding extra definite articles, 10.2% for mistakes in concatenating sentences, 5.2% for wrong WSD of verbs, 3.9% for wrong translation of Persian compound verbs and 2.5% for spelling errors in verb conjugation.

7 Conclusion

PEnTrans is a bidirectional machine translation project which aims to translate between Persian and English.

PEnT1 is the English to Persian side of PEnTrans. It translates standalone English sentences regardless of the context. It combined rule-based method with semantic approaches to improve the result. A novel extension for Lesk algorithm is introduced. PEnT1 uses WordNet glosses and hierarchy, eXtended WordNet to extract WSD tags which makes the proposed work unique. The evaluation results show the superiority to available systems. PEnT2 is the Persian to English side of PEnTrans. It exploits the grammatical roles of each sentence words as the main clue to perform translation. The result shows admissible performance and accuracy.

Covering more complex sentences, supplementing the rules to cover some more special cases and producing a complete syntactically tagged corpus are some of defined further works for this project.

References

- Agirre, E. & Martinez, D. 2001. *Knowledge sources for word sense disambiguation*. In Proc. of International Conference on Text, Speech and Dialogue TSD'2001) Selezna Ruda, Czech Republic.
- Amtrup, W. Mansouri Rad, H. Megerdoomian, K. and Zajac, R. 2000. *Persian-English Machine Translation: An Overview of the Shiraz Project*. New Mexico: NMSU. CRL. Memoranda in Computer and Cognitive Science (MCCS-00-319).
- Anvari, H and gazarani, H. 2004. *Sokhan Children Dictionary*. Sokhan Pub. Teheran. Iran.
- Banerjee, S. and T. Pedersen 2002. *An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet*. Lecture Notes In Computer Science, Berlin / Heidelberg, Springer.
- Christopher D. Manning, D. Klein 2003. *Accurate Unlexicalized Parsing*, 41st Meeting of the ACL.
- Christopher D. Manning, D. K. 2003. *Fast Exact Inference with a Factored Model for Natural Language Parsing*. Advances in Neural Information Processing Systems 15 (NIPS 2002), Cambridge, MA, MIT Press
- Feili, H. and G. Ghassem-Sani 2004. *An Application of Lexicalized Grammars in English-Persian Translation*. Proceedings of the 16th European Conference on Artificial Intelligence (ECAI 2004), Universidad Politecnica de Valencia, Valencia, Spain.
- Feili, H. and G. Ghassem-Sani 2004. *Using Tree Adjoining Grammar in English-Persian Translation*. 9th Annual Int. CSI Computer Conference (CSICC'2004). Sharif University of Technology, Tehran, Iran.
- Feili, H. and G. Ghassem-Sani 2005. *Using a Decision Tree Approach for Ambiguity Resolution in Machine Translation*. 10th Annual Int. CSI Computer Conference (CSICC'2005). Tehran, Iran.
- Fellbaum, C. 1998. *WordNet An Electronic Lexical Database*, MIT Press.
- Ramakrishnan G., Prithviraj B. P., Bhattacharayya P., 2004. *A Gloss Centered Algorithm for Word Sense Disambiguation*. SENSEVAL International Workshop On the Evaluation of System for the Semantic Analysis of Text, Barcelona, Spain.
- Hahn, W. 2003. *Knowledge representation in Machine Translation*. 8th Int. Workshop on Parsing Technologies, IWPT-2003. pp 91-103.
- Harabagiu, S. M., G. A. Miller, et al. 1999. *Wordnet 2 - a morphologically and semantically enhanced resource*, University of Maryland, SIGLEX-99.
- Kiani, S. 2009, *Persian text tokenization and chunking*, 14th International CSI Computer Conference. Tehran. Iran.
- Lesk, M. 1986. *Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone*. SIGDOC '86: the 5th annual international conference on Systems documentation, New York, USA, ACM Press.
- Mahootian, Sh. *Persian Routledge*. 1997.
- Marneffe, M.-C. d., B. Maccartney., Manning, C.. 2006. *Generating Typed Dependency Parses from Phrase Structure Parses*, Italy-Genoa, LREC.
- Montoyo, A. Su'arez A. Rigau G. and Palomar M. 2005. *Combining Knowledge- and Corpus-based Word-Sense-Disambiguation Methods*. Journal of Artificial Intelligence Research 23. 299-330.
- Mosavi, T. and A. Delavar 2005. *Word Sense Disambiguation Using Target Language Corpus in a Machine Translation System*, Literary and Linguistic Computing: 237-249.
- M. Rouhizadeh, M. Shamsfard and M. A. Yarmohamadi, 2008, *Building a WordNet for Persian Verbs*, 4th Global WordNet conference (GWC 2008), Szeged, Hungary.
- Wilks, Y. and Nirenburg, S. 2009. *Machine Translation*. Springer.
- Zhou, X. and Han, H. 2005. *Survey of Word Sense Disambiguation Approaches*. FLAIRS.