Discourse-aware Statistical Machine Translation as a Context-Sensitive Spell Checker

Behzad Mirzababaei, Heshaam Faili and Nava Ehsan
School of Electrical and Computer Engineering
College of Engineering
University of Tehran, Tehran, Iran
{b.mirzababaei,hfaili,n.ehsan}@ut.ac.ir

Abstract

Real-word errors or context sensitive spelling errors, are misspelled words that have been wrongly converted into another word of vocabulary. One way to detect and correct real-word errors is using Statistical Machine Translation (SMT), which translates a text containing some real-word errors into a correct text of the same language. In this paper, we improve the results of mentioned SMT system by employing some discourse-aware features into a log-linear reranking method. Our experiments on a real-world test data in Persian show an improvement of about 9.5% and 8.5% in the recall of detection and correction respectively. Other experiments on standard English test sets also show considerable improvement of real-word checking results.

1 Introduction

Kukich (1992) has categorized errors of a text into five categories: 1. isolated error 2. syntactic error 3. real-word error 4. discourse structure and 5. pragmatic error. In this paper, we focus on the third category, which is also referred as context-sensitive spelling error. This type of error includes misspelled words that are converted to another word of the dictionary (e.g., typing "arm" instead of "are" in the sentence "we arm good"). In order to detect and correct this kind of error, context analysis of the text is crucial.

Here, we propose a language-independent method, which is based on a phrase-based Statistical Machine Translation (SMT). In this case, the input and output sentences are both in the same language and the input sentence contains some real-word errors.

Phrase-based SMT is weak in handling long-distance dependencies between the sentence words. In order to capture this kind of dependencies, which affects detecting the correct candidate word, mentioned SMT is augmented with a discourse-aware reranking method for reranking the N-best results of SMT.

Our work can be regarded as an extension of the method introduced by Ehsan and Faili (2013), in which they use SMT to detect and correct the spelling errors of a document. But here, we use the N-best results of SMT as a candidate list for each erroneous word and rerank the list by using a discourse-aware reranking system which is just a log-linear ranker.

Shortly, the contributions of this paper can be summarized as follow: The N-best results of SMT are regarded as a candidate list of suspicious word, which is reranked by using a discourse-aware reranking system. Two discourse-aware features are employed in a log-linear ranker. The keywords in whole document surrounding the erroneous sentence are considered as the context window. We have achieved about 5% improvement over the SMT-based approach in detection and correction recall and 1% in precision on English experiment. The state-of-the-art results are achieved for Persian context-sensitive spell checker respect to F-measure and Mean Reciprocal Rank metrics.

This paper is organized as follows: Section 2 presents an overview of related works. In Section 3, we explain attributes of Persian language. In section 4, we will describe how to use SMT for generating candidate words. In Section 5, we discuss the approach for reranking the N-best result of SMT. Finally, we illustrate the experimental results and compare the results with the SMT-based approach.

Proceedings of Recent Advances in Natural Language Processing, pages 475–482,
Hissar, Bulgaria, 7-13 September 2013.

2 Related Works

Most of the previous works in real-word error detection and correction are classified into two categories : 1. based-on statistical approaches (Bassil & Alwani, 2012 and 2. based-on separate resource such as WordNet (Fellbaum, 2010) in (Pedler, 2007). Statistical methods use several features, such as N-gram models (Bassil & Alwani, 2012; Islam & Inkpen, 2009), POS tagging (Golding & Schabes, 1996), Bayesian classifiers (Gale, Church, & Yarowsky, 1992), decision lists (Yarowsky, 1994), Bayesian hybrid method (Golding, 1995), latent semantic analysis (Jones & Martin, 1997). The N-gram and POS-based method are combined by Golding and Schabes (1996) and a better result achieved.

Pedler (2007) used WordNet as a separate resource to extract the semantic relations of the words. These methods consider fixed-length windows instead of the whole sentence as the context window.

Most of these methods use confusion set for detecting real-word errors. The confusion set is a set of words that are confusable with the headword of the set. The words of the set are not necessarily confusable with each other (Faili, 2010). When the error checker comes across one of the words in a confusion set, it should select an appropriate word in the sentence. A machine-learning method and the Winnow algorithm is proposed in (Golding & Roth, 1999), to solve word disambiguities based-on surrounding words of the spelling errors. This method uses several features of surrounding words, such as POS tag. +/-10 words from the corresponding confusable word in confusion set are considered as the context window.

Wilcox-O'Hearn et al. (2008) report a reconsideration of the work of (Mays et al., 1991). They use three different lengths for the context window. Also, they use 6, 10 and 14 words as the context window and accommodate all the trigrams that overlap with the words in the window.

Some statistical methods use Google Web 1T N-gram data set to detect and select the best correct word for a real-word error (Bassil & Alwani, 2012; Islam & Inkpen, 2009). Google Web 1T N-gram consists of N-gram word sequences, extracted from the World Wide Web. 5-gram and 3-gram are used in these papers, thus the context window in these methods is 9 and 5 words respectively.

There are few spell checkers for Persian, such as the works presented by Ehsan and Faili (2013); Kashefi, Minaei-Bidgoli, and Sharifi (2010). In Kashefi et al. (2010), a new metric based-on string distance for Persian is presented to rank spelling suggestions. This ranking is based-on the effect of keyboard layout or on the typographical spelling errors.

A language-independent approach based on a SMT framework is presented by (Ehsan & Faili, 2013). This method achieved the state-of-the-art results for grammar checking and context-sensitive spell checking for Persian language. Here, we also use SMT as a candidate generator for spell checking of real word errors, but our approach is different from that work in the following causes: we consider the keywords of whole document as the context-aware features. SMT is used as a candidate generator. We train a log-linear reranking system as a post-processing system to rerank the candidate list.

Our experiments on a real-world test data in Persian show an improvement of about 9.5% and 8.5% in the recall of detection and correction respectively over the method of Ehsan and Faili (2013).

3 Persian Language

Persian or Farsi is an Indo-European language. It is mostly spoken in Iran, Afghanistan and Tajikistan with dialects Farsi, Dari and Tajik respectively. The Persian language has a rich morphology (Megerdoomian, 2000) in which words can be combined with a very large number of affixes. Combination, derivation, and inflection rules in Persian are uncertain (Lazard & Lyon, 1992; Mahootian, 2003).

The alphabet of Farsi is the same as Arabic with four additional letters. The alphabet contains 26 consonants and 6 vowels. Also there are some homophone and homograph letters. For example, "ز", "ذ", "ظ" and "ض" are homophones which all sound as "/*z*" and "ب"/*b*, "پ"/*p*, "ت"/t and "ث"/s are homograph letters which just differ in number and place of dots. These phonetic and graphical similarities cause many spelling errors. In the next section, we will describe how to use the SMT to detect context-sensitive spelling errors in a sentence and generate candidates.

4 SMT as a Candidate generator

SMT framework can be used to model context-sensitive spell checker, which translates a word that does not fit in a sentence with some

suggestions for the suspicious word. SMT uses parallel corpora as the training data. It learns phrases of the language and some features such as phrase probability, reordering probability. In order to use SMT framework, a confusion set for each word is defined. Confusion set of a headword, w_i is a set of words $\{w_{i1}, w_{i2}, \ldots, w_{in}\}$, in which each word w_{ij} is a word that could be converted to w_i with one editing operation of insertion, deletion, substitution or transposition.

The Damerau-Levenshtein distance metric (Damerau, 1964) has been used for calculating the distance between two words. If their distance is lower than a pre-defined threshold, one editing operation, two words have been considered similar and then w_j is added to the confusion set of w_i. For example, confusable words in confusion set of the word روز ruz 'day' are as follows: روزه ruze 'fast', روش ravesh 'method', رود rud 'river', روح ruh 'spirit'.

If $E=\{w_1, w_2, \ldots, w_i, \ldots, w_n\}$ is a sentence and w_i is a real-word error in the sentence, it could appear in several confusion sets, thus, there are several headwords as candidates for the suspicious word. In other words, each headword that has w_i in its confusion set can be suggested as the correct word. To formulate this, consider $C=\{w_1, w_2, \ldots, w_i', \ldots, w_n\}$ is the correct sentence then w_i' is defined as follows (Ehsan & Faili, 2013):

$$w_i' = w_i \, or \, (w_{j,0} \; such \; that \; \exists_{j,k} w_{j,k} = w_i) \qquad (1)$$

Equation (1) implies that the correct word, w_i', is either w_i or one of the headwords that contain w_i. For each erroneous sentence E, which contains real-word error w_i, we can define the N-best candidate sentences $\hat{C}$ as follows:

$$\hat{C} = N - argmax_C \frac{P(E|C)P(C)}{P(E)} \qquad (2)$$

P(E) in Equation (2) is probability of occurring the erroneous sentence, which is constant for each candidate sentence and can be removed from Equation (2). *P(E|C)* can be defined as follows:

$$P(E|C) = P(w_1, \ldots, w_i, \ldots, w_n | w_1, \ldots, w_i', \ldots, w_n) \quad (3)$$

In Equation (3), each w is a word. In order to estimate $P(E|C)$ in Equation (3) we can convert E and C from word base to phrase base, $E = \bar{e}_1, \bar{e}_2, \ldots, \bar{e}_I$ and $C = \bar{c}_1, \bar{c}_2, \ldots, \bar{c}_I$. Using phrase-based SMT, we can capture some local dependencies among the words resulting better detection and correction on real-word errors. Let assume that w_i is in j-th phrase of E, then, we can estimate $P(E|C)$ as follows:

$$P(E|C) = P(\bar{e}_j \mid \bar{c}_j) = \frac{count\ (\bar{e}_j, \bar{c}_j)}{\sum_{\bar{e}_j} count\ (\bar{e}_j, \bar{c}_j)} \qquad (4)$$

Equation (4) is the same as phrasal translation model in phrasal SMT systems. Therefore, we can use a phrasal SMT to correct context-sensitive spelling errors. In this paper, Moses (Koehn et al., 2007) is used as the phrasal SMT.

When using SMT as a context-sensitive spell checker, source and target sentences are in same language. The source sentences contain real-word error while the target sentences contain their correct form. After generating candidate sentences by retrieving the N-best results of the mentioned SMT, we rerank the candidate list by discourse-aware features, which are described in next section.

5 Discourse-aware Features

For any given sentence, SMT-based approach retrieves a list of candidate sentences. The phrasal SMT does not take the whole context of the sentence into account. Thus, in order to find the correct sentence from the candidate list and obtain a better ranking, we define other features that indicate the affinity of each word in candidate sentences with the whole context. Both the sentence and the whole document are considered as the context of the candidate sentences.

For example in the sentence: "This cat is black.", both "cat" and "car" could be meaningful. In this sentence, by considering just the sentence as context window, we cannot identify whether "cat" is correct or "car".

Discourse analysis may help us to detect the best candidate. If we know the document is about automobile or animal, then we can have better reranking on candidates. In other word, considering whole document as the context window is more helpful than considering just whole sentence for reranking the candidate.

Here, we get the benefit from discourse by capturing the relations among the words in a candidate sentence and with the keywords of whole document. In Subsection 5.1, we show that by selecting Point-wise Mutual Information (PMI) measure, we can find the long distance dependency between the words in a document.

Candidate	1 st	2 nd	3 rd	4 th	5 th	6 th	7 th
Detected word	دندان=>چندان	دندان=>دندان	دندان=>زندان	دندان=>مندان	**دندان=>دزدان**	متر=>مصر	دندان=>بندان
$PMI_{sentence}$	-10.8908	-10.8103	-10.8506	-10.9654	**-9.94**	-10.7639	-10.8488
$PMI_{discourse}$	-7.1539	-7.1549	-7.1548	-7.1552	**-7.05**	-7.1606	-7.1523

Table 1: One erroneous sentence with 7 candidate sentences and their PMIs.

5.1 Contextual Features

We select some features that describe the information about the context of the sentences. PMI is used to measure the relation between candidate sentences and the document; and also to measure the co-occurrence among words of the sentence. Another feature that gives us useful information about fluency of candidate sentences is language model (LM) of sentence. A monolingual corpus is required to calculating PMI and LM. PMI of two words of *A* and *B* is calculated as follows:

$$PMI(A,B) = \frac{Doc\_Count\ (A,B)}{Doc\_Count\ (A) \times Doc\_Count\ (B)} \qquad (5)$$

In Equation (5), *Doc_Count(A)* is number of documents that contain word A. *Doc_Count(A,B)* is number of documents that contain both *A*, *B*. We formulate two criteria based on PMI for each candidate sentence $PMI_{discourse}$ and $PMI_{sentence}$. $PMI_{discourse}$ is the PMI of the candidate sentence with its discourse while $PMI_{sentence}$ is the PMI of words candidate sentence. PMI for all words of the candidate sentence with the keywords of document is calculated as $PMI_{discourse}$. For extracting the keywords, term frequency (TF) and inverse document frequency (IDF) measure is like (Li & Zhang, 2007). For each sentence of the test data, 50 keywords are extracted from its discourse. To formulate this, consider *W* as a sentence in the test data and $S_j=\{w_{j1},w_{j2},\ldots,w_{jn}\}$ as j-th candidate sentence resulted from SMT-based approach. Let $C_w=\{c_1,c_2,\ldots,c_{50}\}$ is 50 keywords of the document containing W. $PMI_{discourse}$ for S_j is calculated as follow:

$$PMI_{discourse}\ (S_j) = \frac{\sum_{k=1}^{n}\sum_{m=1}^{50} \mathrm{PMI}\,(w_{jk};c_m)}{n*50} \qquad (6)$$

In Equation (6), n is the number of sentence words. c_m is the m-th keyword of discourse and w_{jk} is k-th word of j-th candidate for *W*. Since PMI measures the co-occurrence of two different words, two identical words has maximum PMI in the sentence. In this case, if a word in the candidate is a keyword of the context, corresponding $PMI_{discourse}$ is increased. Consider S_j={This,cat,is,black} and S_k={This,car,is,black} are candidates of erroneous sentence of *W*. If discourse of *W* is about automobile then $PMI_{discourse}(S_k) > PMI_{discourse}(S_j)$, because the co-occurrence of "car" with the keywords of automobile related document is greater than the co-occurrence of "cat" with that keywords.

Second criterion is $PMI_{sentence}$, which refers to co-occurrence of sentence words with each other. To calculate $PMI_{sentence}$, the PMI of all words of the candidate sentence is calculated. To formulate this, consider $S_j=\{w_{j1},w_{j2},\ldots,w_{jn}\}$ is j-th candidate sentence for test sentence W. $PMI_{sentence}$ of S_j is calculated as follow:

$$PMI_{sentence}\ (S_j) = \frac{\sum_{k=1}^{n}\sum_{m=k}^{n} \mathrm{PMI}\,(w_{jk};w_{jm})}{n*\frac{(n-1)}{2}} \qquad (7)$$

In Equation (7), n is number of words of the sentence and w_{jk} is k-th word of j-th candidate of *W*. Table 1 shows an example of our Persian artificial test data in which $PMI_{discourse}$ and $PMI_{sentence}$ of correct candidate are more than that of SMT-based approach suggests. The input sentence is:

دندان قوي هيكل دو متر از ريل راه آهن اوكراين را دزديدند

dandaan-ghavi-hikal-dv-mtr-az-ril-raah-aahan-avkraain-raa-dozdidand

'Robust teeth stole two meters of railway of Ukrainian'.

There are two confusable words in the sentence, دندان dandaan 'teeth' and متر metr 'meter'. SMT generate 7 candidate sentences in which the 5th candidate is the correct one. As shown in Table 1, the first candidate, generated by SMT, has $PMI_{discourse}$ and $PMI_{sentence}$ score less than the correct sentence. By reranking SMT results using $PMI_{discourse}$ and $PMI_{sentence}$, we can put the correct sentence at better rank or the top of the list. The third contextual feature is LM, which is used to score the fluency of the candidate.

We consider surrounding words of suspicious word, whole sentence and whole document as the context, then, we use LM, $PMI_{sentence}$ and $PMI_{sentence}$ to extract information. After calculating $PMI_{sentence}$, $PMI_{discourse}$ and LM for all candidate sentences, a log-linear model is used to rerank the N-best results.

For reranking with log-linear model we need the weight of each feature. Support Vector Machine [1] (SVM) (Tsochantaridis, Joachims, Hofmann, Altun, & Singer, 2006) is used to weight each feature. SVM is a machine-learning algorithm based on statistical learning theory. It has been widely used, especially in function regression (Jeng, 2005) and pattern recognition (Tsai, 2005), in recent years for its better generalization performance (Burges, 1998).

5.2 Feature Weighting

Log linear model is used to rerank the N-best results of SMT. Like (Hayashi, Watanabe, Tsukada, & Isozaki, 2009), we use SVM-rank to obtain the weight of each feature. A corpus contains erroneous and correct sentence is developed. For each sentence of the corpus, $PMI_{sentence}$, $PMI_{discourse}$ and LM is calculated. We use the corpus a training data for SVM-rank to obtain the weight. In next section, the details of all data sets are described more precisely.

6 Experiment Result

We evaluate the accuracy of the approach by using the false positive and false negative rates as follows: *False positive* (FP) errors refer to real-word errors that were not identified by SMT-based system. *False negative* (FN) errors refer to appropriately written word that SMT-based approach detected as real-word error. *True positive* (TP) results are correct words that are considered as correct. *True negative* (TN) results refer to real-word errors that SMT-based approach detected and changed regardless of the correction. Finally *True negative with correction* (TNC) are real-word errors that SMT-based approach was able to replace them with the correct word. Evaluation metrics are computed as follows:

$$Precision = \frac{\#\ of\ TNC}{\#\ of\ TN} \quad (8)$$

$$Correction\ Recall = \frac{\#\ of\ TNC}{\#\ of\ FP\ and\ TN} \quad (9)$$

$$Detection\ Recall = \frac{\#\ of\ TN}{\#\ of\ FP\ and\ TN} \quad (10)$$

Another metric for evaluating our N-best result retrieved by SMT, is Mean Reciprocal rank. It is calculated as follows:

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i} \quad (11)$$

In Equation (11), $|Q|$ is the number of sentences of test data and $rank_i$ is the rank of correct sentence in 20-best result. We tested the SMT-based approach on two different languages, English and Persian. In the next subsections, we illustrate results on Persian and English languages.

6.1 Results on Persian Language

Our train data is generated from Peykareh (Bijankhan, 2004), Hamshahri[2] and IRNA[3] data sets. Hamshahri and IRNA are collections of news documents of Persian language. These corpora contain 814, 166,774 and 179,574 documents of general texts respectively. They have 56,241, 576,137 and 332,343 types and 2,530,772, 78,841,045 and 64,085,181 tokens respectively. All three corpora contain 923,744 types.

Our confusion set is generated from all mentioned data sets. It includes 5,000 headwords and each headword has about 4 confusable words in average. For our experiments on Persian, we have deployed two different test sets: an artificial and a real-world test sets.

Our Persian real-world test data for context-sensitive spelling errors contains 1,100 sentences. The test set selected manually from the Internet mostly from Persian weblogs[4]. Each sentence contains 16.7 words in average and only one real-word error. The test set contains 27 insertion errors, 266 deletion errors, 527 substitution errors and 91 transpositions errors. Only 89 errors, 8% of whole errors, need more than one editing action.

We also made an artificial test data for context-sensitive spelling errors. 1,500 sentences were selected randomly from Peykareh corpus. Length of each sentence is between 4 and 20 words. For each sentence in the artificial test set, one real-word error was inserted artificially, by replacing a random word with a word in its confusion set.

Our training corpus contains 381,007 sentence pairs which are selected form mentioned corpora. After generating training data, Moses is used as our SMT system, GIZA++ (Och & Ney, 2003) is used for word alignment and SRILM (Stolcke, 2002) is used as LM toolkit. Our LM is created from Hamshahri and IRNA and contains 329,607

[1] http://svmlight.joachims.org/

[2] The Hamshahri2 test collection is available on: http://ece.ut.ac.ir/DBRG/Hamshahri/.

[3] Islamic Republic News Agency-http://www.irna.ir

[4] The test set is available on: ece.ut.ac.ir/nlp/resources/

unigrams, 4,764,131 bigrams and 6,228,300 trigrams.

In order to develop training data for SVM, a confusion set is generated. The confusion set contains 26,891 headwords, which are selected from Hamshahri and Peykareh. Each headword has 4.6 confusable words.

5,000 sentences from Hamshahri and Peykareh are selected randomly. All sentences have at least one headword in the confusion set. For each sentence, one word of the sentence is selected and replaced with one of its headword. For each erroneous sentence maximum 20 candidates are generated by SMT. 56,320 sentences are generated and 3,728 of them are correct sentences. For each sentence of training data, $PMI_{sentence}$, $PMI_{discourse}$ and LM are calculated and their values normalized. We used 56,320 sentences as training data for SVM-rank to obtain the weights.

We generate a candidate list for each sentence of test sets by using the SMT and rerank the list in a post-processing step. In Table 2, results of discourse-aware reranking on real-world and artificial test data are shown. We selected the work of Ehsan and Faili (2013) as a baseline.

Experiments on Persian	Artificial test data	Real-world test data
Precision	0.97(-0.01%)	0.83(-0.01%)
Detection recall	0.70(+16%)	0.73(+9.5%)
Correction recall	0.69(+15%)	0.61(+8.4%)
F-measure	0.80(+8.4%)	0.70(+4.4%)
MRR	0.71(+8%)	0.67(+4%)

Table 2: Summarized results on Persian test sets (the improvements are mentioned in parentheses).

As it is shown in Table 2, in both test sets, the proposed ranker retrieved a significant superior result over the baseline with respect to recall metric with a comparable precision. Since the principle of discourse-aware SMT is language independent, we tested it on English language too.

6.2 Results on English Language

The test sets for English language were drawn from two corpora: Wall Street Journal (WSJ) and Brown corpus. For WSJ test set, a confusion set is generated with 73,437 headwords and each headword has 5.9 confusable words in average. We extract confusable words from WSJ based on one editing action. 1,500 sentences are selected from WSJ randomly similar to the test sets developed in (Islam & Inkpen, 2009; Wilcox-O'Hearn et al., 2008). For each sentence, a real-word error is inserted randomly. Rest of WSJ is considered as training data for SMT.

Similar work of Golding and Roth (1999); Jones and Martin (1997), we use 20% Brown corpus as test data and apply on 19 confusion sets. The test data contains 3015 erroneous sentences [1]. Train data for SMT, is generated from WSJ and rest of Brown corpus, 80%.

We have tested SMT based approach on both artificial English test data, generated candidates and reranked them with discourse-aware features. Table 3 shows results of discourse-aware.

Experiments on English	WSJ test data	Brown test data
Precision	0.97(+0.001)	0.96(+0.008%)
Detection recall	0.90(+5.4%)	0.81(+2.6%)
Correction recall	0.87(+5.6%)	0.78(+3.2%)
F-measure	0.92(+3%)	0.86(+2.1%)
MRR	0.88(+3%)	0.83(+1%)

Table 3: Summarized results on English test sets (the improvements are mentioned in parentheses).

As shown in Table 3, in WSJ and Brown test sets, our proposed system outperforms the baseline with respect to all metrics. We have a significant improvement over the baseline with respect to detection and correction recall.

7 Conclusion & Future work

We improved SMT-based approach by extracting some contextual features and using a learning algorithm, SVM-rank, for getting weights of each feature and reranking the N-best results by a log-linear model. The proposed ranker retrieved a significant superior result over the baseline with respect to recall metric with a comparable precision.

Real-word errors with two editing actions can be injected to training data. An ontology, named FarsNet (Shamsfard, 2008), can be used as an external resource to identify Persian semantic relationships between words. We can use discourse-aware reranking as a Learning To Rank, and apply it on every method that generate N-best result.

[1] The test set is available on: http://cogcomp.cs.illinois.edu/Data/Spell/

References

Bassil, Youssef, & Alwani, Mohammad. (2012). Context-sensitive Spelling Correction Using Google Web 1T 5-Gram Information. *arXiv preprint arXiv:1204.5852*.

Bijankhan, Mahmood. (2004). The role of the corpus in writing a grammar: An introduction to a software. *Iranian Journal of Linguistics, 19*(2).

Burges, Christopher JC. (1998). A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery, 2*(2), 121-167.

Damerau, Fred J. (1964). A technique for computer detection and correction of spelling errors. *Communications of the ACM, 7*(3), 171-176.

Ehsan, Nava, & Faili, Heshaam. (2013). Grammatical and context‐sensitive error correction using a statistical machine translation framework. *Software: Practice and Experience, 43*(2), 187-206.

Faili, Heshaam. (2010). *Detection and correction of real-word spelling errors in Persian language.* Paper presented at the Natural Language Processing and Knowledge Engineering (NLP-KE), 2010 International Conference on.

Fellbaum, Christiane. (2010). WordNet: an electronic lexical database. *WordNet is available from http://www. cogsci. princeton. edu/wn*.

Gale, William A, Church, Kenneth W, & Yarowsky, David. (1992). A method for disambiguating word senses in a large corpus. *Computers and the Humanities, 26*(5-6), 415-439.

Golding, Andrew R. (1995). *A Bayesian hybrid method for context-sensitive spelling correction.* Paper presented at the Proceedings of the third workshop on Very Large Corpora.

Golding, Andrew R, & Roth, Dan. (1999). A winnow-based approach to context-sensitive spelling correction. *Machine learning, 34*(1-3), 107-130.

Golding, Andrew R, & Schabes, Yves. (1996). *Combining trigram-based and feature-based methods for context-sensitive spelling correction.* Paper presented at the In Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics, Santa Cruz, CA.

Hayashi, Katsuhiko, Watanabe, Taro, Tsukada, Hajime, & Isozaki, Hideki. (2009). Structural support vector machines for log-linear approach in statistical machine translation. *Proceedings of IWSLT, Tokyo, Japan*.

Islam, Aminul, & Inkpen, Diana. (2009). *Real-word spelling correction using Google Web IT 3-grams.* Paper presented at the Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3-Volume 3.

Jeng, Jin-Tsong. (2005). Hybrid approach of selecting hyperparameters of support vector machine for regression. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 36*(3), 699-709.

Jones, Michael P, & Martin, James H. (1997). *Contextual spelling correction using latent semantic analysis.* Paper presented at the Proceedings of the fifth conference on Applied natural language processing.

Kashefi, O, Minaei-Bidgoli, B, & Sharifi, M. (2010). A novel string distance metric for ranking Persian spelling error corrections. *Language Resource and Evaluation*.

Koehn, Philipp, Hoang, Hieu, Birch, Alexandra, Callison-Burch, Chris, Federico, Marcello, Bertoldi, Nicola, . . . Zens, Richard. (2007). *Moses: Open source toolkit for statistical machine translation.* Paper presented at the Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions.

Kukich, Karen. (1992). Techniques for automatically correcting words in text. *ACM Computing Surveys (CSUR), 24*(4), 377-439.

Lazard, Gilbert, & Lyon, Shirley A. (1992). *A grammar of contemporary Persian*: Mazda Publishers.

Li, Juanzi, & Zhang, Kuo. (2007). Keyword extraction based on tf/idf for Chinese news document. *Wuhan University Journal of Natural Sciences, 12*(5), 917-921.

Mahootian, Shahrzad. (2003). *Persian*: Routledge.

Mays, Eric, Damerau, Fred J, & Mercer, Robert L. (1991). Context based spelling correction. *Information Processing & Management, 27*(5), 517-522.

Megerdoomian, Karine. (2000). *Unification-based Persian morphology.* Paper presented at the Proceedings of CICLing.

Och, Franz Josef, & Ney, Hermann. (2003). A systematic comparison of various statistical alignment models. *Computational linguistics, 29*(1), 19-51.

Pedler, Jennifer. (2007). Computer correction of real-word spelling errors in dyslexic text. *Unpublished PhD thesis. Birkbeck, University of London*.

Shamsfard, Mehrnoush. (2008). *Developing FarsNet: A lexical ontology for Persian.* Paper presented at the 4th Global WordNet Conference, Szeged, Hungary.

Stolcke, Andreas. (2002). *SRILM-an extensible language modeling toolkit.* Paper presented at the Proceedings of the international conference on spoken language processing.

Tsai, Chih-Fong. (2005). Training support vector machines based on stacked generalization for image classification. *Neurocomputing, 64*, 497-503.

Tsochantaridis, Ioannis, Joachims, Thorsten, Hofmann, Thomas, Altun, Yasemin, & Singer, Yoram. (2006). Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research, 6*(2), 1453.

Wilcox-O'Hearn, Amber, Hirst, Graeme, & Budanitsky, Alexander. (2008). Real-word spelling correction with trigrams: A reconsideration of the Mays, Damerau, and Mercer model *Computational Linguistics and Intelligent Text Processing* (pp. 605-616): Springer.

Yarowsky, David. (1994). *Decision lists for lexical ambiguity resolution: Application to accent restoration in Spanish and French.* Paper presented at the Proceedings of the 32nd annual meeting on Association for Computational Linguistics.