IJICT International Journal of Information & Communication Technology

ITRC

Volume 2- Number 2- July 2010 (pp.51-59)

Research Note (IT Section)

The First Persian Context Sensitive Spell Checker

Heshaam Faili
School of Electrical and Computer Engineering,
College of Engineering, University of Tehran,
Tehran, Iran
hfaili@ut.ac.ir

Mohammad Azadnia
Information Technology Department
Iran Telecom Research Center
Tehran, Iran
azadnia@itrc.ac.ir

Received: February 28, 2009- Accepted: July 28, 2010

Abstract: **In this article, an attempt to introduce the first Persian context sensitive spell checker, which tries to detect and correct the real-word spelling error of Persian text is presented. The proposed method is a statistical approach which uses Bayesian framework as its probabilistic model and also uses mutual information metric as a semantic relatedness measure between different Persian words. Our experiments on sample test data, shows that accuracy of correction method is about 80% with respect to F1-measure.**

Keywords- Context Sensitive Spell Checker; Persian language; real-word error; Bayesian framework, mutual information

I. INTRODUCTION

Real-word spelling errors occur when the intended word of the writer is misspelled and the resulting error is a valid word in the lexicon. These errors can happen due to typing mistakes or due to "Auto-Correction" feature of word-processing software (Wilcox-O'Hearn et al., 2008).

Kukich (1992) categorized the spelling text errors into five types. The first type is the lexical errors, which include non-word errors. For example the word "حمله/hamleh/attack" maybe mistakenly typed by non-word "خمله/khamleh/".

Most of these errors could easily be found by conventional spell checkers. The second type is the syntactic errors, which result in ungrammatical text. The detection and correction of such errors requires a grammar checker. For example the verb in the following sentence are not appropriately inflected based on the its subject:

(1) man be khaneh raft.
I to home went(he/she/it)

The third kind is the semantic errors which appear in the text when the writer misspells a word for another word that can take the same grammatical role. As an example, the word "حمله/hamleh/attack"can be typed wrongly to word"جمله/jomleh/sentence".

They cannot be detected by conventional spell checkers and grammar checkers. These types of error are usually incongruous to the surrounding text. In this paper, this kind of errors in Persian texts is considered. The final two types of error hierarchy are discourse structure and pragmatic errors, which cannot be classified as spelling errors.

Real-word errors mostly falls into the third level of Kukich's categories but it can also causes syntactic anomaly, which is related to the second type of the mentioned errors and can be detected by a grammar checker. Most conventional spell checkers are unable to detect such errors as they only check each word to see if they can be found in the dictionary or not. The misspelled word which should be a non-word error is flagged as an error. But detection of real-word errors requires some context-sensitive analysis of the text to obtain knowledge about the intention of the writer (Golding and Roth, 1999). While a human reader may easily detect such errors and suggest reasonable corrections, automatic detection of real-word errors can be very tricky. Some of these errors can go undetected even by human readers (Hirst and Budanitsky, 2005).

The whole previous methods on real-word errors detection and correction can be divided into two distinct approaches: based on a separate resource and based on machine learning and statistical methods (Wilcox-O'Hearn et al., 2008).

Hirst and Budanitsky (2005) proposed a method from the first approach, which uses WordNet as an external resource to detect real-word errors. This approach detects words in the text that are semantically distant from nearby words and if it finds a word that is semantically closer in the context, flags the original word as an error. They tested the method on an artificial corpus of errors; it achieved the recall between 23–50% and the precision between 18–25%. Similar approach which also uses WordNet as a semantic resource was proposed by (Pedler, 2005). He used WordNet as a resource to find semantic relationships in order to detect real-word errors. The small experiment on using WordNet resource has shown that the proposed method of gathering co-occurring nouns by their WordNet hypernyms could successfully capture the semantic associations of confusable words. This method also used the threshold of confidence level. The maximum threshold 0.99 has obtained one false alarm and just 5% of the errors are corrected. With setting the threshold to 0.9 just over half of the errors are corrected and 14 false alarms remained.

Several machine learning and statistical methods have been used for context-sensitive spelling correction. Atwell and Elliott (1987) addressed the problem by looking for unlikely part-of-speech bigrams. While Mays et al. (1991) frame the problem as a noisy channel problem and combine it with the trigram model to assess the probability that a sentence is correct by considering all variants of the sentence by replacing words with their spelling variation and computing trigram probabilities. Recently, Wilcox-O'Hearn et al. (2008) consider the limitations of Mays et al. (1991) evaluation and reevaluate the method. They show that the later method is superior in performance to the WordNet-based method of Hirst and Budanitsky (2005), especially when supplied with a realistically large trigram model.

Golding and his colleagues (Golding and Roth, 1999; Golding and Schabes, 1996) address the "context-sensitive spelling correction" as a "word disambiguation" problem. Ambiguity among words is modeled by predefined confusion sets. Golding (1995) compared performance of decision lists and Bayesian classifiers. The latter was found to give better performance, especially when it combines with a trigram part-of-speech method (Golding and Schabes, 1996). Golding and Roth (1999) follow this method by applying a Winnow multiplicative weight-updating algorithm to the same problem and retrieved a considerable improvement in accuracy (around 95%). This led to the development of the SNoW (Sparse Network of Winnows) architecture. Carlson et al. (2001) report the achievement of a high level of accuracy (99%) when applying this method to 256 confusion sets. In contrast to these statistical techniques, Mangu and Brill's (1997) propose a transformation-based learning approach which uses far fewer parameters, although it achieved comparable performance.

Also, a context-sensitive spelling correction task is presented in (Ingason, et al, 2009) where adapts established methods from English to a morphologically rich language, Icelandic, and concludes that although rich morphology negatively affects performance, their system is still good enough to be useful in regular word processing.

These statistical methods usually rely on a predefined confusion sets. These sets contain words called confusables that are likely to be mistakenly used in place of another one. On encountering one of the members of the sets in the text, the spell-checker checks to see which of the words in the set are more appropriate in the surrounding context. These methods learn some features in order to distinguish each of the confusable word in a typical context. Based on these features and the surrounding text, each member of the set is scored and the word with the highest score is the most appropriate one in the context.

One method of creating confusion sets, as used by Wilcox-O'Hearn et al. (2008), is to collect words that differ from each other by a single letter (insertion, deletion, substitution or transposition). The confusion set is created of this type for each word in their 20,000 word vocabulary of English. Word trigram probabilities, derived from a large body of text are learned. To simulate error correction, 100 sentences of correctly spelled text (containing only words in their vocabulary) are taken and from these, over 8000 misspelled sentences are generated by successively replacing each word with each member of its associated confusion set. Each of these misspelled sentences contained just one error. The words appearing in the sentence were given a higher probability of their alternative confusion set members to represent the fact that words are more likely to appear correctly spelled than they are to be misspelled. This threshold is called confidence level. Varying the confidence level, they were able to detect 76% of the errors and correct 73% at the expense of just one false alarm and were still able to detect 63% and correct 61% while reducing the false alarms to zero.

When the error checker comes across one of the words in confusion set it should decide which one is more appropriate in the sentence. The statistical methods for finding the best choice are Bayesian classification (Golding, 1995), latent semantic analysis (Jones and Martin, 1997) and winnow (Golding and Roth, 1999). Achieving a high level accuracy of 99% with 256 confusion sets was reported in [51] with applying winnow based approach.

An advantage of statistical over knowledge-based methods is that they can handle function words as well as content words. They can also consider words that are not spelling variation of another one. The drawback of these methods is that they are limited to predefined confusion sets and they check every word in the text that could be found in the confusion sets.

From the best of our knowledge, there is no any published works on context-sensitive Persian error checking system. In this paper, we present a statistical method that basically fits in the second category. This method uses a predefined confusion set that is built by using different heuristic on distance measuring algorithm for Persian dictionary. In the next sections, after introducing general features of Persian language, the generation of the confusion sets in Persian is illustrated. Then, a method that uses "Mutual Information" of pairs of words to score each of the confusable words in order to suggest the possible correction are demonstrated. Finally the evaluation results of the mentioned method are presented.

II. Confusion Sets

Confusion set is a set of words which are considered confusable with the headword of the set, but are not necessarily confusable with each other (Mays et al., 1991). Each word in a Persian dictionary appears in the confusion set of all members of its own confusion set, which means a word can appear in several sets.

In order to generate the confusion sets, the Levenshtein (1966) distance metric has been used. The Levenshtein distance between two words is defined as the minimum number of edits needed to transform from one into another one. The only acceptable edit actions are insertion, deletion, or substitution of a single character. This metric is a measure to accept a word as in the other's confusion set. If this measure is lower than a defined threshold, the two words are considered similar and each of them is added to the confusion set of the other pair word. The sets obtained in this way are not general enough. Special issues for Persian alphabet must be taken into account.

In Persian, there are letters that are read exactly the same but written differently. These letters are the most common causes of spelling errors. Another category that can cause problems is those that are written similarly but read differently. These letters are more likely to be mistaken for each other while typing. A large number of words in Persian texts are inflective and cannot be found in the Persian dictionaries. Caution taken into consideration is that such inflected words could be in homographic relation with a dictionary entry. Finally, letters that appear adjacent on a standard keyboard can be mistakenly typed in place of each other.

The Levenshtein metric can be modified to give smaller cost to substitutions of letters spoken identically or similarly, or written similarly, or appear adjacent on a standard keyboard. This means that these substitutions are more likely to be happened and the resulting words are more likely to be confused with another one.

Another point that should be taken into consideration is that the first letter of the intended word is less likely to be typed incorrectly. Thus, words that differ in the first letter can be considered to be less likely to be confused.

Based on these assumptions, the confusion sets have been generated. The sets can be pruned further, if a grammar-checker is used. If a word is replaced with another word that always has a different part-of-speech, i.e. different grammatical role in the sentence, then the appropriate word can be found using a grammar checker. Thus the words that have different part-of-speech, compared to the headword in a set, can be removed. The drawback to the use of confusion sets is that it limits our real-word error correction to a set of predefined limited number of words.

Table 1. Persian Confusion Set Statistics

Number of confusion sets	1165535
Average number of confusion set members	8.7
Number of unique words	1187981
Number of words which has no any confusion pair	22446

The confusion set of any Persian word is collected by computing the mentioned distance between the head word and any other Persian word and accepting as a confusable pairs for small value distance. By setting the maximum edit distance of confusable words to be at most one operation and limiting the maximum number of words in any confusion set to be at most 40, the whole confusion sets of Persian words are generated. Table 1 shows different statistics of these sets.

III. Persian Real-word Error Correction

Based on the Bayesian approach which was previously published by (Gale et al., 1994; Golding, 1995) a method for correcting real-word errors is presented in this section. The method uses a predefined confusion set for each Persian word and a list containing the mutual information of every pair of words in a Persian lexicon as a source of knowledge for its scoring process.

The method relies on predefined confusion sets. A confusion set for the word w_i is a set $\left\{w_{i1}, w_{i2}, \ldots, w_{in_i}\right\}$ that contains the words that are likely to be confused with word w_i, which is called headword of the set.

On encountering a headword in the text, the corresponding set is checked to find the most appropriate word in the context. Like the previous works (Gale et al., 1994; Golding, 1995) the real-word error correction problem is defined as follows:

For an input sentence containing an occurrence of the word w_i , a word w from the set S={ w_i }$\cup$

$\{w_{i1}, w_{i2}, \ldots, w_{in_j}\}$ that maximizes the following probability (equation 1), given the context words c_j observed within a ±k-word window of the target word w_i, is selected. The set $\{w_{i1}, w_{i2}, \ldots, w_{in_j}\}$ are the words of confusion set of headword w_i:

$$W = \arg\max_W \ p(w|c_{-k}, \ldots, c_{-1}, c_1, \ldots c_k) \quad (1)$$

Using Bayes rule, the mentioned probability can be changed to the following equation:

$$p(w|c_{-k}, \ldots, c_{-1}\, c_1, \ldots, c_k)$$

$$= \frac{p(c_{-k}, \ldots, c_{-1}\, c_1, \ldots, c_k|w)p(w)}{p(c_{-k}, \ldots, c_{-1}\, c_1, \ldots, c_k)} \quad (2)$$

The probability $p(c_{-k}, \ldots, c_{-1}\, c_1, \ldots, c_k)$ is the same for all w S because it depends on the words of the input string (i.e. word context) and not on the confusable words. So it can be ignored from equation (1) and is not needed for the comparison between the words of S. p(w) can be computed by Maximum Likelihood Estimation (MLE) method, as a ratio of the total occurrence of word w in the sentences of a corpus to the total number of the sentences in the corpus. Thus, equation (3) is concluded:

$$p(w) = \frac{total\ occurence\ of\ w\ in\ the\ corpus}{total\ number\ of\ sentences\ in\ the\ corpus} \quad (3)$$

The probability $p(|c_{-k}, \ldots, c_{-1}\, c_1, \ldots, c_k|w)$ is almost impossible to be computed from the training data. This probability can be simplified by using the independence assumption of presence of a word in the sentence with any other one. The relaxed computation of the mentioned probability can be written as follow:

$$p(|c_{-k}, \ldots, c_{-1}\, c_1, \ldots, c_k|w) =$$

$$\prod\nolimits_{j \in -k, \ldots, -1, 1, \ldots, k} P(c_j|w) \quad (4)$$

But, the simplest way to compute the probability $p(c_j|w)$, is:

$$p(c_j|w) = \frac{p(c_j \cap w)}{p(w)}$$

$$= \frac{\dfrac{total\ cooccurenc\ e\ of\ c_j\ and\ w\ in\ the\ corpus}{total\ nimber\ of\ sentences\ in\ the\ corpus}}{\dfrac{total\ cooccurenc\ e\ of\ w\ in\ the\ corpus}{total\ nimber\ of\ sentences\ in\ the\ corpus}}$$

$$= \frac{total\ cooccurence\ of\ c_j\ and\ w\ in\ the\ corpus}{total\ cooccurence\ of\ w\ in\ the\ corpu} \quad (5)$$

This estimate might be inaccurate due to the lack of enough training data.

Gale et al. (1994) have used interpolation of $p(c_j|w)$ and $p(c_j)$ to address the problem of lacking adequate training data. The probability $p(c_j|w)$ is the desired one but is subject to inaccuracy because of insufficient training data. The probability $p(c_j)$ is more accurate but might be irrelevant to the desired value $p(c_j|w)$. By interpolation of the two probabilities, they have tried to minimize the inaccuracy of their estimate.

Golding (1995) has addressed this problem by not considering all words in the ±k-words window. If there is not enough training data for a given word c_j to accurately estimate $p(c_j|w)$ for all w, then c_j is simply disregarded, and the discrimination will be based on other, more reliable evidence. They implemented this idea by proposing a "minimum occurrences" threshold, $T_{\min}$, as the threshold to accept a word as a reliable appearance. A context word c is ignored if the following condition happens:

$$\sum\nolimits_{1sisn} m_i < T_{\min} \ \ or \sum\nolimits_{1sisn} (M_i - m_i) < T_{\min} \quad (6)$$

Where M_i is the total number of occurrences of i-th word (w) in the training corpus, and m_i is the number of such occurrences for which c occurred within ±k words window. In other words, c is ignored if it practically never occurs within the context of any w, or if it practically always occurs within the context of every w. In the former case, there is insufficient data to measure its presence; but in the latter, its absence couldn't be estimated.

A context word might also be ignored if it does not help discriminate among the words in the confusion set. Stop-words and most of the function words fall into this category. To determine whether a context word c is a useful discriminator, a chi-square test to check for an association between the presence of c and the choice of word in the confusion set can be used. If the observed association is not judged to be significant, then c is discarded.

In order to estimate the probability $p(c_j|w)$ more accurate, an idea similar to the one which was mentioned by Gale et al. (1994) is proposed. In fact, instead of computing the probability $p(c_j|w)$ directly, mutual information of two words c_j and w, is estimated. Using the equation (7), the desired quantity of equation (1) can be re-written as follow:

$$W = \arg\max_W \ p(w|c_{-k}, \ldots, c_{-1}, c_1, \ldots c_k)$$

$$= \arg\max_w \prod_{j\in -k,\dots,-1,\dots,k} p(c_j|w).p(w)$$

$$= \arg\max_w \sum_{j\in -k,\dots,-1,1,\dots,k} \log p(c_j|w) + \log p(w) \qquad (7)$$

But from the definition of mutual information (see next sub-section), the following equation can be derived:

$$\log p(c_j|w) = MI(c_j, w) + \log p(w) \qquad (8)$$

Where $MI(c_j, w)$ stands for mutual information between c_i and w and p(w) stands for the probability of appearance of word w. From equation (7) and (8), the following equation can be inferred:

$$w = \arg\max_w \qquad (9)$$

$$\left(\sum_{j\in -k,\dots,-1,1,\dots,k} MI(c_j, w)\right) + (2k+1).\log p(w)$$

In the above formula, k stands for the window size which shows the number of neighboring words which its mutual information is considered in the formula.

In order to tackle the problem of data sparseness in computing the mutual information, a similar approach of Golding (1995) for pruning the indiscriminative words by analyzing their mutual information is used. A context word c is ignored if the following condition is satisfied:

$$MI(c,w) < MI_{\min}\ or\ (1 - MI(c,w)) < MI_{\min} \qquad (10)$$

In this case of ignoring a context word c from, the window-size of equation (9) should to be decreased one unit. The mutual information approach proves to yield acceptable result in detecting and correcting real-word errors in Persian.

A. *Persian Mutual Information*

The mutual information of two discrete random variables X and Y is defined as:

$$MI(X,Y) = \sum_{y\in Y} \sum_{x\in X} p(x,y) \log\left(\frac{p(x,y)}{p(x)p(y)}\right) \qquad (11)$$

where p(x,y) is the joint probability distribution function of X and Y, and p(x) and p(y) are the marginal probability distribution functions of X and Y respectively.

In our case, the mutual information of two Persian words w_1 and w_2 is calculated from a corpus collected from the 200,000 articles gathered from IRNA[1] and 1,900,000 sentences borrowed from Hamshahri2[2] as the training data. IRNA is a news agency published their news on different languages, mainly on Persian. Hamshahri is one of the most popular daily newspapers in Iran that has been publishing for more than 20 years. Hamshahri2 corpus is a Persian test collection that consists of 1.4 GB of news texts from this newspaper since 1996 to 2007. From this collection, the news from 1996 to 2002 that contains about 1,900,000 sentences is selected as a part of training corpus.

The mutual information of two words w_1 and w_2 is defined as follow (Church and Hanks, 1989):

$$MI(w_1, w_2) = \log\left(\frac{p(x,y)}{p(x)p(y)}\right)$$

$$= \log\left(\frac{\frac{N(w_1,w_2)}{N}}{\frac{N(w_1)}{N}\frac{N(w_2)}{N}}\right) = \log\left(N \times \frac{N(w_1,w_2)}{N(w_1)N(w_2)}\right) \qquad (12)$$

Where N(x) is the total number of occurrence of x in the corpus and N(x,y) is the number of co-occurrences of x and y in the corpus and N is the size of the corpus. N(x,y) can be measured using a fixed length window. It is estimated by counting the number of times that x is followed by y in a k-words window. It also can be measured as the number of co-occurrences of the words x and y in a sentence as in our case. Another way to measure N(x,y) is the number of co-occurrences of x and y in a news body which can be applied to our case with IRNA and Hamshahri articles. The mutual information of every two words in the corpus is computed using the above formula if the two words satisfy the condition introduced by Golding (1995).

As mentioned before, the highly inflection feature of Persian language can inflate the data sparseness problem in computing the mutual information. Thus, a simple stemming system which groups together the different surfaces of a word is applied before this computation. That is, different inflected forms of a form are regarded as one word in this process. This stemming process is very essential especially when it deals with Persian verbs.

B. *Discussion on the Method*

Based on the equation (1), the problem of scoring the relatedness of ambiguous word, is reduced to finding the w S that maximizes the following term which is called Score(w):

Score(w)=

$$\left(\sum_{j\in -k,\dots,-1,1,\dots,k} MI(c_j, w)\right) + (2k+1).\log p(w) \qquad (13)$$

The Score(w) from the above equation is a good estimation for measure of relatedness word w, because on average, choosing the most frequent word yields a successful result in 74.8% of the cases used by Golding and Roth (1999) as the base line method.

[1] Islamic Republic News Agency (http://www.irna.ir)

[2] The Hamshahri2 test collection is made available for download at http://ece.ut.ac.ir/DBRG/Hamshahri/.

Mutual information of w and c reflects their co-occurrences with respect to their occurrence in the training corpus. That is, if the total co-occurrences of w and c, and the occurrence of w on the training data are multiplied by d > 1, the mutual information would not change:

$$MI(w,c) = \log(\frac{N(w,c)}{N(w)\times N(c)} =$$

$$\log\left(\frac{d\times N(w,c)}{(d\times N(w))\times N(c)}\right) \qquad (14)$$

In the proposed method, a word w gets a higher score, if it has a higher occurrence in the training data. If there are other real-word errors within a ±k-word window of the target word w_i the method would not make a good decision, especially if there are fewer discriminating words within the window. This is because Score(w) depends on the mutual information of w and also depends on the other real-word errors which may exist in the window as well as on other words. This would results in an unreliable score for w. If there is, within the ±k-word window of the target word w_i, a word from the confusion set of w_i or even w_i itself again, then the method can make unreliable decisions. The mutual information of a word with itself (i.e. entropy of the word) is not a good measure in our case because the co-occurrence of a word with itself is usually low in the training data and a reliable result cannot be obtained. But in the above case, this mutual information would be needed and if the word turns out to be the intended word of the writer, then the score for the intended word would be lower than expected which is not what we want. This can be avoided by not considering the words in the window that come in the confusion set of w_i or w_i itself but the final scores would then be less reliable.

IV. Evaluation

In order to evaluate the proposed approach, confusion sets of every Persian word are needed. Also, the mutual information of every pair of Persian words should be estimated to compute the score of each confusable word. The mentioned IRNA and Hamshahri articles have been used for computing the mutual information of Persian words and testing the real-word error detection and correction approach.

For testing the method, 100 sentences shorter than 15 words, apart from training articles are chosen from IRNA corpus. Then, one of the words in each sentence was randomly replaced with a randomly chosen word from its confusion set to obtain a ‘corrupted’ version as a test sentence. In the whole sentences only one real-word error are generated. Also, those sentences don’t contain any other word from the confusion set of the target word. In each sentence the noisy word are flagged as a suspicious word. The algorithm should correct this word and propose a correct candidate in each of the mentioned sentence.

In all experiments, the size of window is set to the maximum length of the sentences. So, the whole words of the sentences fall into the word-window. From the whole words of any sentence, only those that satisfy the condition of equation (10) are considered in the scoring process.

To find possible candidates for each real-word error, the correction algorithm needs to scan the text and check every word in the text with all the headwords of the confusion sets. If it encounters a match, the method should be applied to the word to check if it is a real word error.

A. *Evaluation Results*

The experiments on real-word error detection and correction algorithm can be divided into two different categories: completeness and soundness evaluation. To evaluate the completeness of the algorithm, the noisy test data are fed into the system to be corrected. While to evaluate the soundness, the gold data are used to be checked. In each sentence of both corpora, one of the words is flagged as a suspicious word[3]. The mentioned algorithm checks the suspicious word as a candidate for the real-word error and all the words in the confusion set of the candidates are scored differently.

In the experiments for evaluating the soundness of the algorithm, which the errorless data is fed to the system as a test data, the following two different outcomes may be resulted:

Accept Correctly: In this case, the suspicious words are detected as no-error correctly.

Flag as Error: In this case, the suspicious word is flagged as an error incorrectly and a suggestion which is in fact not the intended word of the writer is made.

But, in the second experiments, which a noisy data is fed to the system as a test data, the following three different results may happen:

Flag as Error and make the correct suggestion: In this case, the suspicious word is flagged as an error and the intended word of the writer is made as a suggestion successfully.

Flag as Error and make an incorrect suggestion: the suspicious word is flagged as an error but a word other than the intended word of the writer is made as a suggestion. It means that the detection phase is correctly resulted but the correction fails to propose the desired word.

Ignore: In this case, the noisy word could not be detected by the system and it’s ignored.

Table 2, shows the evaluation results of both experiments. As shown in the table, our method gets the 79% performance both in detecting the error and non-error correctly.

Table 2. The evaluation results on Soundness and Completeness Evaluation Experiments

[3] The suspicious words in the noisy test data are those that are changed to an error word, but in the gold data these word are chosen randomly.

Soundness Evaluation		Completeness Evaluation		
Accept	Flag	Flag& correct	Flag not correct	Ignore
79%	21%	79%	8%	13%

By combining the mentioned experiments, the precision and recall of real word error detection method are estimated as follows:

Precision=

$$= \frac{Number\ of\ correctly\ detected\ errors}{total\ number\ of\ detected\ errors} = \frac{87}{87+21} = 80.50\%$$

$$Recall = \frac{Number\ of\ correctly\ detected\ errors}{total\ number\ of\ existed\ errors} = \frac{87}{100} = 87\%$$

B. Discussion

From the whole possible results of the algorithm, accept correctly in the case of soundness evaluation and flagging as an error and making the correct suggestion in the case of the completeness evaluation are the desired outcomes of the system. The Algorithm yields identical outcomes for both evaluation categories, considering other results as wrong decisions in the case of the completeness evaluation.

Flagging as error in soundness evaluation, is the most undesirable output as it means that the writer's intended word in the text is flagged as error. This happens in 21% of the cases. Usually this happens when the scores for some of the words in the confusion sets and the intended word of the writer are so close but the intended word which is in the text has a lower score.

By default, the algorithm selects the word with the highest score as the correct word. In fact, the word w from the confusion set is selected when its score is greater than others (i.e. the following condition holds):

$\forall$ x $\in$ S=$\{W_i\}\cup\{W_{i1},W_{i2},\ldots,W_{in_i}\}$ and x $\neq$ w,
Score(w) > Score(x) (15)

where **W**$_i$ is the target word and the headword of the confusion set $\{W_{i1},W_{i2},\ldots,W_{in_i}\}$.

If a coefficient d > 1 is added to introduce a confidence level to the mentioned condition, then many of the false alarms can be avoided. This coefficient shows our confidence level to current written word. The equation (16) shows the modified condition for accepting a candidate as a real-word error:

$\forall$ x $\in$ S = $\{W_i\}\cup\{W_{i1},W_{i2},\ldots,W_{in_i}\}$ & x $\neq$ w,
Score(w) > d × Score(x) (16)

This condition means that the score for w must be higher the maximum score for other words multiplied by coefficient d.

The coefficient can vary from very close to one (e.g. 1.1) or a very large number like 100. The higher value of d, the more false alarms is avoided, but more performance is loosed in the completeness evaluation. This performance is in fact the intended goal of this method.

Table 3, shows the results of real-word error correction and detection algorithm on different values of confidence level.

Table 3. The accuracy of the algorithm with different confidence levels

	soundness evaluation		completeness evaluation	
Confidence Level (d)	Accept	Flag	Flag & Correct	Ignore or Incorrect suggestion
1	79%	21%	79%	21%
1.2	82%	18%	78%	22%
1.7	84%	16%	72%	28%
2	86%	14%	70%	30%
4	87%	15%	51%	49%
10	90%	10	40%	60%
100	99%	1%	18%	82%

By defining a measure to combine the completeness and soundness metric, the best value on confidence level can be achieved. Here a weighted average of the mentioned two measures which is called Fβ-measure is used. The Fβ-measure is often used in the field of information retrieval for measuring search, document classification, and query classification performance [Steven M. Beitzel. (2006)]. Earlier works focused primarily on the F1-measure, but with the proliferation of large scale search engines, performance goals changed to place more emphasis on either precision or recall and so Fβ is seen in wide application. Equation (17) shows this combination measure.

$$F_\beta = \frac{(1+\beta^2).(Precision.Recall)}{(\beta^2.Precision+Recall)} \quad (17)$$

The performance of completeness evaluation as a Precision and the performance of soundness evaluation as a recall measure are used in the mentioned formula.

Figure 1, shows the best resulted **F**$_\beta$of the mentioned experiments for different values of β. In each value of β, the confidence value in which, the best result is achieved is shown too.

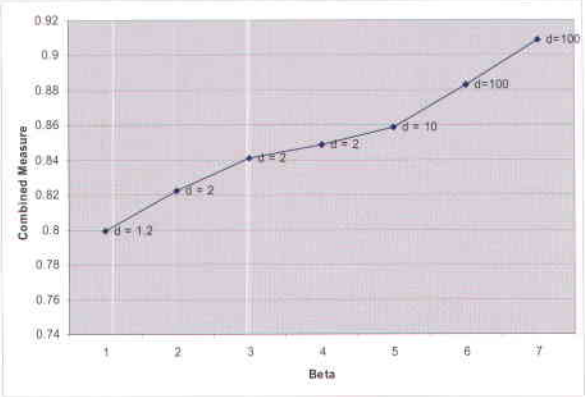

Figure 1. The best value of F_β in which the algorithm achieves for different values of β

V. Conclusion

We have presented a method for context sensitive spelling correction in Persian language. In order to find possible candidates for real-word error, we have generated a confusion set, a set of words that are likely to be confused with a certain word, for every word in the training corpus. Then based on the Bayesian framework of (Gale et al., 1994), we have proposed a method using mutual information of words to score words including the target word and the words in its confusion set. Based on this score, the algorithm selects the word with the highest score and gives an outcome.

We have achieved acceptable performance considering the limitations caused by insufficient training data and some corruption introduced to the data of the corpus. The suggested improvements using confidence levels can help to obtain better result for practical usage.

The test set of the system was generated by injecting just one real-word error in each Persian sentence. The context of each word is defined to be the whole sentence words. Thus, this assumption that each sentence only contains at most one real-word error is equal to assuming that the whole context of noisy word is correct. Although, we couldn't experiment this assumption, but it seems that increasing the number of real-word errors in each sentence, affects on the quality of error detection and correction dramatically. For example, suppose that the sentence (1) was mistyped to sentence (2):

(1) "روز/rooz/day" "روشن/roshan/bright" "است/ast/is"

The day is bright.

(2) "رود/rood/river" "روان/ravan/flowed" "است/ast/is"

The river is flowed.

The word "روز/rooz/day" was mistyped to another word "رود/rood/river" and "روشن/roshan/bright" is mistyped to "روان/ravan/flowed". The mutual information of both word in both sentences is high and the system couldn't detect the error due to consistency in context words.

VI. References

[1] Atwell, E. and Elliott, S. 1987. Dealing with ill-formed English text, In Garside, R., Leech, G. and Sampson, G., editors, The Computational Analysis of English: A Corpus-Based Approach, 120–138, Longman.

[2] Carlson, A.J., Rosen, J. and Roth, D. 2001. Scaling Up Context Sensitive Text Correction, Proceedings of the National Conference on Innovative Applications of Artificial Intelligence 45–50.

[3] Church, K., Hanks, P. 1989. Word Association Norms, Mutual Information, and Lexicography, In Proceedings of the 27th Annual Conference of the Association of Computational Linguistics.

[4] Gale, William A. and Church, Kenneth W. 1994 What's Wrong with Adding One? In Oostdijk, N and de Haan, P. (Eds.) Corpus-based Research into Language. pp. 189-198. Rodopi, Amsterdam.

[5] Golding, A.R. 1995. A Bayesian Hybrid Method for Context-sensitive Spelling Correction, Proceedings of the Third Workshop on Very Large Corpora, 39-53.

[6] Golding, A.R. and Roth, D. 1999. A Winnow based approach to context-sensitive spelling correction, Machine Learning 34, 107-30.

[7] Golding, A.R. and Schabes, Y. 1996. Combining Trigram-based and Feature-based Methods for Context sensitive Spelling Correction, Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics, 71-78.

[8] Hirst, G and Budanitsky, A. 2005. Correcting Real-Word Spelling Errors by Restoring Lexical Cohesion, Natural Language Engineering, vol. 1, No. 11, 87-111.

[9] Ingason, A.K., Jóhannsson, S.B., ögnvaldsson, E. R, Helgadóttir, S., Loftsson, H., , 2009, Context-sensitive spelling correction and rich morphology, Proceedings of NODALIDA.

[10] Jones M.P., Martin, J.H., 1997, Contextual spelling correction using latent semantic analysis, Proceedings of the fifth conference on Applied natural language processing, p. 173.

[11] Kukich, K. 1992. Techniques for Automatically Correcting Words in Text, Computing Surveys vol. 24, No. 4, 377- 439.

[12] Levenshtein VI. 1966. Binary codes capable of correcting deletions, insertions, and reversals, Soviet Physics Doklady, No. 10, 707–10.

[13] Mays, E., Damerau, F.J. and Mercer, R.L. 1991. Context Based Spelling Correction, Information Processing and Management, vol. 25, No. 5, 517-22.

[14] Mangu, L. and Brill, E. 1997. *Automatic Rule Acquisition for Spelling Correction,* Proceedings of the 14th International Conference on Machine Learning (ICML 97), 187-194.

[15] Pedler, J. 2005. *Using semantic associations for the detection of real-word spelling errors,* In Proceedings from The Corpus Linguistics Conference Series, vol. 1, no. 1, Corpus Linguistics.

[16] Steven Beitzel M. 2006. *On Understanding and Classifying Web Queries,* Phd Thesis, http://ir.iit.edu/~steve/beitzel_phd_thesis.pdf.

[17] Wilcox-O'Hearn, A., Hirst, G., and Budanitsky, A. 2008. *Real-Word Spelling Correction with Trigrams: A Reconsideration of the Mays, Damerau, and Mercer Model,* In Proceedings of 9th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing-2008).

Heshaam Faili received his B.S and M.S degree in Software Engineering and his Ph.D in Artificial Intelligence from Sharif University of Technology on 1997, 1999 and 2006 respectively. He joined to the Artificial Intelligence and Robotic group of department of electrical and computer engineering in University of Tehran on 2008 and still working in this group as a member of faculty. The main research interest is Natural Language Processing (NLP) mainly on Persian, such as Machine Translation, Spell Checking, Langauge learning, classificaiton and clustering systems. His major approaches are Statistial and Machine Learning based and sometimes an hybrid model of rule-based and probabilistic methods also been used.

Mohammad Azadnia received his B.S degree in Telecommunication Engineering from Iran University of Science and Technology in 1988, and his M.S. degree in Industrial Management from Sharif University of Technology Tehran, Iran. He has been working in Iran Telecommunication Research Center as a Researcher and Project Manager since 1988. He continued his activity as a Member of faculty in IT department of ITRC since 1999. He has published more than 35 papers in international conference and journals. His research interests are NLP, IR, IS and IT Management.