Context's impact on the automatic spelling correction

Mohammed Nejja*

TSE Team,
ENSIAS Mohammed V University,
Rabat, Morocco
Email: mohammed.nejja@gmail.com
*Corresponding author

Abdellah Yousfi

Eradiass Team,
FSJES Mohammed V University,
Rabat, Morocco
Email: yousfi24qma@yahoo.fr

Abstract: This paper aims to shed light on a mechanism that will be used to exploit topical context information improving the accuracy of the automatic spell checking system. This study aims at solving the problem encountered by the auto correct spell checking system, which resides in the fact that the requested solution might be located in the last position. We have implemented a set of techniques in order to build a context oriented spelling corrector and by the end of this work the designed corrector will essentially be using a dictionary that contains a distribution of probability of a word occurrence in various contexts. This latter is constructed by bringing a collection of documents available via the internet.

Keywords: Arabic natural language processing; ANLP; automatic spell checker; context; N-gram; Levenshtein.

Reference to this paper should be made as follows: Nejja, M. and Yousfi, A. (2017) 'Context's impact on the automatic spelling correction', *Int. J. Artificial Intelligence and Soft Computing*, Vol. 6, No. 1, pp.56–74.

Biographical notes: Mohammed Nejja is a PhD student at ENSIAS. He is currently preparing his PhD in third year. The research theme is about the automatic spilling correction in Arabic language.

Abdellah Yousfi holds a Doctorate in Automatic Speech Recognition in 2001. In 2003, he was integrated as a Professor at Mohamed V University, Rabat, Morocco. He is also a member in the research team telecommunications and embedded system (ENSIAS) and responsible of the research axis: the natural language processing (in particular of the Arabic language). He is interested in the following research areas: automatic speech recognition of the Arabic language, the morphological and syntactic analysis of the Arabic language. Information retrieval and the automatic correction of spelling errors.

This paper is a revised and expanded version of a paper entitled 'The context in automatic spell correction' presented at The International Conference on Advanced Wireless, Information, and Communication Technologies, Tunisia, 5–7 October 2015.

1 Introduction

The automatic language processing (ALP) is a discipline that combines linguists and computer scientists closely. The ALP is designed to model and develop computer solutions to handle linguistic data automatically based on rules that are represented in some calculable formalism. The main areas of the ALP are: text processing, speech processing, automatic translation, text indexing and retrieval of information from the internet and voice control of domestic robots. Numerous works have been developed in this area for languages such as English and French. Today, morphological, syntactic and semantic analyses are part of the essential concerns of researchers working in this field. For the Arabic language which is yet part of the four most widely-spoken languages in the world, we are still far from those results. But it has to be noted that several research teams are working on the subject of the automatic correction, morphological and syntactic analysis and that several works have already been developed or are being developed.

In this work, we will focus on the automatic spelling correction systems according to the context without considering the grammatical mistakes. That means that the spelling checker, as we consider it, will not detect an error in the sentence **خرجت الاستاذ من القسم** (Xarajati Al>usotadu mina Alqisomi|the teacher left the class). These kinds of faults need to have morphological information and syntactic rules about each word in the sentence. In fact, the automatic correction techniques are considered to be among the first techniques developed in the natural language processing (NLP) field.

The researches on the automatic verification and correction debuted in the USA around the late '50s and continue nowadays. In the field of automatic spell check, we distinguish between two steps:

- error detection: consists of verifying the appearance of a word in the language vocabulary which requires a large lexis to cover all the words in the language
- error correction: consists of suggesting to the user word replacements that are lexically relative to the word entered.

In this paper, we offer a method of automatic spelling correction considering the context. This correction aims to organise the solutions proposed by the corrector and to efficiently identify the most suitable solution in the given context.

The present paper is structured as follows. The second section presents an art state for the algorithms and techniques destined for the automatic spelling correction. We will be focusing on the automatic spelling correction in the Arabic language, considering that it is our research topic. The third section is devoted to the description of the automatic spelling correcting according to the context, mainly its characteristics; the Arabic language features and our proposed algorithm to solve the issue of the automatic spelling correction according to the context for the Arabic language. The fourth section will be

introducing the developed application to assist the end user efficiently dealing with texts written in Arabic. We conclude the article by offering a set of perspectives concerning the improvement of the developed tool.

2 The Arabic language

The Arabic language is an Eastern Semitic language which is written and read from right to left. Its alphabet has 28 consonants adopting different spellings depending on their position (at the beginning, in the middle or at the end of a lexical unit) and eight diacritics. Arabic is the fourth spoken language in the world (Nwesri et al., 2005; Abdelhadi et al., 2007).

2.1 Characteristics of the Arabic language

2.1.1 Vowelisation

An Arabic lexical unit is written with consonants and vowels. The vowels are added above or below the letters. They are required to properly read and understand a text; they also allow to differentiate the lexical units having the same representation.

2.1.2 Flexion

A fusional language is a language in which the lexical units vary in number and flexion (either the number of names, or verbal time) following grammatical relations they have with other lexical units.

2.1.3 Agglutination

Agglutination is the addition of 'proclitic' called prefixes and suffix so-called 'enclitic' simple forms to obtain what it calls 'agglutinative forms'. The proclitic and the enclitic form all of the enclinomenes of the Arabic language.

2.2 Essential elements of the Arabic language

2.2.1 Root

Roots are at the origin of most Arabic words. They are verbs consisting of three to five consonant letters (Mustafa et al., 2008). A root defines the fundamental meaning of the derivative words using different diacritics and affixes with the letters of the root to create an inflexion of the meaning (Wehr, 1961).

2.2.2 Models or patterns

The Arab model (patterns) essentially allows to determine the structure of most words (names, verbs, etc.). The models are variations of the word فعل (faEala – to do) which are obtained by using diacritics or by adding affixes.

2.2.3 Affixes

Affixes are letters added at the beginning (prefix) or at the end of Arabic words (suffixes). In general, they are used to give the words syntactic elements.

2.3 Particularities and difficulties

The Arabic language has specificities related to its morphology, syntax and semantic which make it more ambiguous than other natural languages. Among the particularities of this language.

2.3.1 Correlation

The agreement in gender and number between nouns and adjectives or even the agreement between nouns and verbs creates a strong correlation between the different levels of the language processing namely, morphological, syntactic and semantic. This correlation depends on the animation and/or the nature of the name in question. If we consider, for example, the phrase **العصافير نائمة** (AlEaSaAfiyru naAJmap – the birds are sleeping), we find that the word العصافير (AlEaSaAfiyru-Birds) which is masculineplural is not consistent in number and gender with the singular feminine adjective **نائمة** (naAJmap – are sleeping) as it is inhumane. This is not the case with the الأطفال نيام (Al>aTofaAlu niyaAm – kids are sleeping) sentence where the agreement between the name and the word exists, as **الأطفال** (Al>aTofaAlu – kids) is a human animated name, and so the adjective نيام (niyaAm – are sleeping) is the masculine plural.

2.3.2 The absence of vowels

Most of Arabic documents are not vowelised. Indeed, the Arabic alphabet consists only of consonants and each consonant can have different pronunciation and meanings depending on the diacritics used, particularly in the case of the ذهب (*ahaba) word which can be seen as the name ذهب (*adabN – the gold) which means gold or the verb ذهب (*ahaba – to go) which means leave. It should be noted that multiple emphasis is very common in the Arabic language, given the significant proportions of the ambiguous words.

2.3.3 The irregularity of the order of words in a sentence

In the Arabic language, the order of the Arabic word is relatively free. We can change the order of words in a sentence and obtain sentences having the same meaning. For example:

- أخرج المخرج فيلما جديدا (>axoraja Almuxoriju filomFA jadidFA – the filmmaker produced a new movie)
- أخرج فيلما جديدا المخرج (Almuxoriju >axoraja filomFA jadidFA – the filmmaker produced a new movie)
- فيلما جديدا أخرج المخرج. (filomFA jadidFA >axoraja Almuxoriju – the filmmaker produced a new movie).

This order causes ambiguities.

2.3.4 Text segmentation

Segmentation of texts is based on the topical context exploration of punctuation and words connectors playing the role of separator of sentences لقد (laqad – to have), لكن (lakin – but) and أمّا (>ama A – as for) as well as those of some particles such as conjunctions of coordination و (wa – and) and ف (fa-). For example: الى القسمو القى الدرس ذهب الاستاذ (ahaba Al<usotaA*u <laY Alqisomi wa <alqaY Aldarosa* – the teacher went to the class and conducted the lesson) In this sentence, the particle و (wa-and) plays the role of separator between two proposals الذهاب الى القسم (Al*ahaAb<laY Alqisomi – going to class) and القاء الدرس (<iloqaA' daros – conducting the lesson). On the other hand, in the following sentence ذهب الاستاذ و المدير الى القسم (ahabaAl < usotaA * uwaAlmudiru < laYAlqisomi – the teacher and the school director went to the class*) the same particle و (wa – and) does not play the role of separator between proposals but rather that of a conjunctive between the words الاستاذ (Al<usotaA*u – the teacher) and المدير (Almudiru – the director) hence does not segment the sentence.

2.3.5 Problems of agglutination of words

The conjunctions of coordination play an important role; however, they stick to the words that follow. Thus for example the letter و (wa) in the word وهم (wahom – imagination) may represent a letter of the word in question 'imagination' or a combination of a personal pronoun و+ هم (wa hum – and they) followed by coordination.

3 State-of-the-art

The automatic spelling correction is a major task for a set of text analysis systems. The main purpose is to have an automatic spelling checker that is able to spot and correct the misspelled words while keeping the lowest execution time. In this section, we briefly introduce the different techniques used for an automatic spelling correction.

Thanks to the efforts of many researchers in the field of automatic spelling correction, numerous techniques and algorithms appeared. Among those researches, we note:

- The work of Kukich (1992), Mitton (1996) and Peterson (1980) which is a classic algorithm based on the word search in a dictionary. If that researched word does not appear in the dictionary, then it is said that it is erroneous. This latter technique has the main disadvantage of having a prohibitive execution and dictionary data access time. Faced with this issue, Knuth (1973) has introduced the idea of hash table that allows a selective access to the researched word.
- The typical approaches based on the minimal editing distance calculation. Those techniques are highly used for the automatic spelling correction. The technique measures the similarity level between two strings. It calculates the minimal number of elementary editing operations to go from one word to another. One of the well known algorithms in this field is the algorithm of Damerau (1964). The algorithm considers a misspelled error as a combination of elementary editing to the insertion; deletion; transposition and the substitution. Based on the algorithm of Damerau (1964), Levenshtein (1966) has defined a new distance that calculate the minimum number of elementary operations to go through to go from one word to another. This

distance is based on only three mistake types which are: the addition, substitution and the deletion of characters.

- The finite-state automaton has also been a research topic for the automatic spelling correction. This technique has been used by Aho and Corasick (1975) and has set an algorithm consisting of going through an abstract structure named dictionary containing the researched words by reading the text letters one by one. The data structure is established due to a sorting or digital tree to which we add suffixes links. A sorting can be seen as transitions function of a deterministic finite-state automaton. Once the dictionary is established, the algorithm has a linear complexity of the text and researched strings length.

- Oflazer (1996) presents a new approach based on the recognition notion with error tolerance thanks to a finite states identificator. This latter is founded on the use of a dictionary constituted with a finite states automaton and a distance said cut-off edit distance.

- Savary (2000) has introduced a correction algorithm founded on the Oflazer's works. The Savary's algorithm is different due to his cut-off edit distance variation. First, it searches the presence of the input word in the automaton. In case of failure, it goes back (back-hacking). Every time it goes back to a previously visited state, it ties to find another continuation for the path, admitting one of the four editing operations (insertion, reversal, omission and replacement).

- Another modelling introduced by Pollock and Zamora (1984) consists in associating an alpha-code to each word of the dictionary (the consonants forming the word), which explains the necessity of having two dictionaries: one for the words and another for their alpha-codes. Consequently, the correction is made by comparing the alpha-codes with the erroneous word. This method is efficient for permutation errors.

- The n-gram approach (Ukkonen, 1992) based on the decomposition of a word in n elements and build with a sequence of data. This technique compares each sequence in a knowledge corpus to produce a similarity indication to indicate the nearest words to the misspelled one. These researches were also focused on several languages including Arabic language.

Thereby, several studies about the correction of Arabic texts were carried out and are available for use, such as:

- Gueddah et al. (2012) suggested a new approach so as to improve planning solutions of an erroneous word in Arabic documents by integrating frequency editing errors matrices in the Levenshtein algorithm.

- A new approach has been advised by Bakkali et al. (2014) based on the use of a dictionary of the stems of Buckwalter to integrate morphological analysis in the Levenshtein algorithm.

- Ben Othmane and Ben Ahmed (2003) proposed a new method aiming to reduce the number of proposals given by automatic Arabic spelling correction tools. They suggest the use of error's context in order to eliminate some correction candidates.

- Nejja and Yousfi (2015) presented a new algorithm based on a dictionary constituted of surface patterns and roots characterised by a scaled down size .Thereby, they identified initially, the nearest surface pattern to the erroneous word. Then, they created a set of potential candidates deriving all the roots according to the identified surface pattern. Finally, they compared the erroneous word with these words by using the edit-distance.

- Attia et al. (2015) developed their dictionary of 9.2 million fully-inflected Arabic words (types) from a morphological transducer and a large corpus, validated and manually revised. They improve the error model by analysing error types and creating an edit distance re-ranker. They also improve the language model by analysing the level of noise in different data sources and selecting an optimal subset to train the system on.

- Shaalan et al. (2012) created an adequate, open-source and large-coverage word list for Arabic containing 9,000,000 fully inflected surface words. Furthermore, from a large list of valid forms and invalid forms they create a character-based tri-gram language model to approximate knowledge about permissible character clusters in Arabic, creating a novel method for detecting spelling errors. Testing of this language model gives a precision of 98.2% at are call of 100%. They take their research a step further by creating a context-independent spelling correction tool using a finite-state automaton that measures the edit distance between input words and candidate corrections, the noisy channel model, and knowledge-based rules. Their system performs significantly better than Hunspell in choosing the best solution, but it is still below the MS Spell Checker.

- Shaalan et al.'s (2003) approach is heuristic and involves developing an Arabic morphological analyser, techniques of spelling checking and spelling correction, and efficient methods of lexicon operations. The developed Arabic spell checker is able to recognise common spelling errors for standard Arabic and Egyptian dialects.

- Hassan et al. (2014) are concerned with the first four error types as they contribute more than 90% of the spelling errors in the corpus. The proposed system has many models to address each error type on its own and then integrating all the models to provide an efficient and robust system that achieves an overall recall of 0.59, precision of 0.58 and F1 score of 0.58 including all the error types on the development set.

These algorithms have proven their effectiveness in the field of automatic correction of spelling. However these approaches do not support the context of the error. As an example, we consider the misspelled word رساك (rasaAk) in the following sentence: نزل جبريل بالوحي على رساك الله صلى الله عليه و سلم وهو في غار حراء (Nazala jiborilu biAlwaHoyi EalaY rasaAko Alla hi salaY Alla hi Ealayohi wa sala m wa huwa fiy gaAri HiraA' – the angel Gabriel has come with the revelation to rasaka at the Hirae cave). We can mention as solutions رسام (rasaAm – designer), رسول (rasuwl – prophet), رسوب (rusuwb – failure), etc. Thus, the nearest proposition based on spelling is رسام (rasaAm – designer). Yet, considering the context of the sentence which is ISLAM, the solution رسول (rasuwl – prophet) is the most suitable.

4 The correction by using the topical context

Topical context correction consists of classifying correctly the suggested solutions based on the general context, paragraph, or words surrounding the error detected while exploiting a set of topical context information to organise the proposed solutions and identify the most adaptable one to the context regardless of its location.

Therefore, the construction of an automatic correction spell system according to the context for the Arabic language must involve different types of information. In this paper, we set up the problem of this type of correction. There are different correction algorithms according to the context:

- The n-gram algorithm: an n-gram is a sequence of n consecutive characters to identify the candidate that looks best in the order of words where spelling errors exist. n-grams models are widely used in many applications. They are generally associated with the recognition of handwritten characters techniques (Srihari and Baltus, 1992), translation word (Brown et al., 1990; Cattoni et al., 2001), recognition of speech (Rosenfeld, 2000), and of course to find real-word errors (Wilcox-O'Hearn et al., 2008). In the field of spelling correction, the model n-gram algorithm provides plenty of information to select the appropriate correction. For a spelling error, several candidates are generated. This spelling error occurs in a sequence of segments of context. The adjacent words to the spelling error can help choose the adequate correction. The calculation considers, for the prediction of a word, that the sequence of $n - 1$ words that precede is sufficient. The basic calculation of these probabilities is therefore made by a count of each sequence observed based on a variety of segments of context, various sizes [which corresponds to so-called models trigrams (Woszczyna, 1998; Johnson et al., 1999; Gauvain et al., 2000; Bacchiani et al., 2001; Jelinek, 2001)], positions (based not only on the words that precede the spelling errors in the text, but also the words that follow) and spanning the misspelled word.
- Word association: an algorithm that organises solutions according to the semantic aspect. It puts the candidate having a better semantic fit with the words around the spelling error. Basically, we can take each candidate and measure its semantic relatedness to check out how it fits with each word in the text. Note that semantic adjustment gives a value of proximity between two meanings of words that appear together often, and which are adjacent in the same perceptual area. In contrary, the semantic similarity which includes words that appear similar, i.e., sharing a number of functional (perceptible) both descriptive properties.
- Repetitions of words: words that occur several times in a text can help find the appropriate corrections. When the same word, or its inflectional variant, is encountered in the text, the candidate is reinforced.
- The subject of text: the organisation of the proposed solutions is made according to the relevance of the candidates for the subject of text. If the subject of a text is known, a thematic word list can be given preferential status during the correction of misspellings for the text. In a document, several topics can be treated that why segmentation becomes an important step to identify the segments containing the information relating to the subject. Segmentation is the process of cutting a text into

smaller units to disambiguate the borders of sentences and paragraphs. Text segmentation is based on the linguistic study on the one hand, and on computer modelling on the other hand. Automatic segmentation of Arabic texts presents several specific difficulties because of the punctuation which is rarely used in Arabic texts and not always determinative to guide the segmentation. Furthermore, non-vowelised Arabic text is highly ambiguous. The proportion of the ambiguous words can go up to 90% if counts relate to vowelisation global (Debili et al., 2002). Thus, a non-vowel word may have several possible morphological features (Chadben and Belguith, 2003). For example the word كتب (kataba – to write) may be a name, a verb, or a pronoun, etc. Arabic is an inflectional language in which words change forms according to their grammatical relationship to other words in a sentence. Thus, the identification of the grammatical category of words is ambiguous causing difficulties at the level of the automatic segmentation. Unlike most Latin languages, Arabic is not supported on the punctuation signs. Thus, throughout an Arabic paragraph we might not find punctuation signs except a period at the end of this paragraph. We also include problems of agglutination of words. In the Arabic language a word can sometimes match any phrase for example أتلعبون (>ataloEabuwna – Do you play?).

4.1 Our solution

The search for solutions to the spelling correction problem in the context of Arabic text remained a challenge for a long time. Several researchers focused on the problem and thanks to their efforts various techniques and algorithms have emerged. For instance, the work of Aouragh et al. (2015) where they have developed a system for correcting spelling errors in the Arabic language based on language models and Levenshtein algorithm. Another modelling conducted by Ben Othmane and Ben Ahmed (2003) which consists of using the reversal rule of Bayes to calculate the probability of the right solution, given the words that surround the error in the text for each candidate. The most successful algorithm to date is Golding and Roth (1999) winnow-based spelling correction algorithm, which is able to recognise about 96% of context-sensitive spelling errors, in addition to ordinary non-word spelling errors. For real-word spelling correction, Fossati and Di Eugenio (2007) have shown the usefulness of parts-of-speech contexts, they have proposed a methodology based on a mixed trigrams language model. Xu et al. (2011) have proposed a novel way of incorporating dependency parse and word co-occurrence information into a state-of-the-art web-scale n-gram model for spelling correction The technique proposed in this paper is aimed at addressing the problem of:

- the absence of a sequence in the corpus of learning (their likelihood is therefore set to 0) but may appear to the user
- the estimation of the number of parameters increases exponentially with the length of the words that adjoin it
- estimation of joint probabilities is practically impossible for the Arabic language due to the grammatical richness of this language.

4.1.1 Learning of the model context

We have extended the knowledge corpus by collecting available documents from Wikipedia (60,237 documents). These documents have been grouped into ten topics, namely: art; geography, history, medicine, politic, religion, sciences, society, sport, technology named $TC_{j(j=1,j=2\ldots10)}$. After downloading and classifying the collected documents, they were thoroughly analysed to select the words to be added in the database. This selection starts at first by a verification phase to check the correspondence between the words constructing a document and the category to which it belongs. The keywords that seem to have a meaning associated to the document's category are selected and adopted in our corpus of learning. The empty words, first name … were ignored. In addition, to construct a rich learning corpus, we downloaded documents containing keywords that correspond to at least two identified categories. Once the keywords constructing our corpus are identified, we calculated the occurrences of each word in the context to determine the relative probability of each word based on the following formula:

$$P\left(w_i/TC_j\right) = \frac{\text{The ocurrences of } w_i \text{ in } TC_j}{\text{total number of } w_i \text{ in our training corpus}}$$

4.1.2 Detection of context

In this step we identify the context of each paragraph. For that we first chunk the paragraph into text using carriage returns (/r), then we filter the text by eliminating punctuation marks and words providing no information but help identifying the context such as pronouns, articles, etc. while keeping the carriers terms of information. After that we identify the corresponding context *TC* for each paragraph (noted *Par*) by using the following formula:

$$TC = \arg\max_{TC_j} \sum_{w_i \in Par} P\left(w_i/TC_j\right) \quad (1)$$

either

- $P(w_i/TC_j)$: relative probability of w_i in the context TC_j, this probability is estimated through the relative frequency calculated on our learning corpora.

4.1.3 Correction

To correct an error based on the context we start by running an overall text analysis to identify the misspelled words. Based on the contexts identified previously, we organise the suggested solutions by putting in first place the most suitable candidate to the context. Note:

We_{rr}	a misspelled word
W_c	desired word
TC	the context where the misspelled word is found and detected by the formula (1)

$W_{pi} = w_{pl1}, w_{pl2}, \ldots W_{pln}\}$ the set of solutions of W_{err} classified by the Levenshtein distance

D_{lev} Levenshtein distance.

To improve the classification of W_{pl}, we propose the following measure noted MC:

$$MC\left(w_{err}, w_{pli}\right) = \frac{D_{lev}\left(w_{pli}, w_{err}\right)}{P\left(w_{pli}/TC\right)+1}$$

4.1.4 The correction process of our approach

The proposed system analyses the text (identify correct words, misspelled words, articles, etc.). If there are any misspelled words, the system chunks the text into paragraph having a definite context, then it matches each misspelled word with its corresponding paragraph, then it suggests the most lexically close candidate for each misspelled word. If the system managed to determine the context of the paragraph containing the misspelled word, it organises proposed solutions based on the context identified. If the system is unable to identify the context based on the correct words, due to either:

- the correct words do not provide information allowing the identification of the context such as verbs, empty words, etc.
- these words do not exist in our learning corpus, it uses the lexically closest candidates such as key words (already having a probability of occurrence in the corpus of learning) to identify the context, then it organises the proposed solutions based on the identified context.

Figure 1 Diagram describing the correction process (see online version for colours)

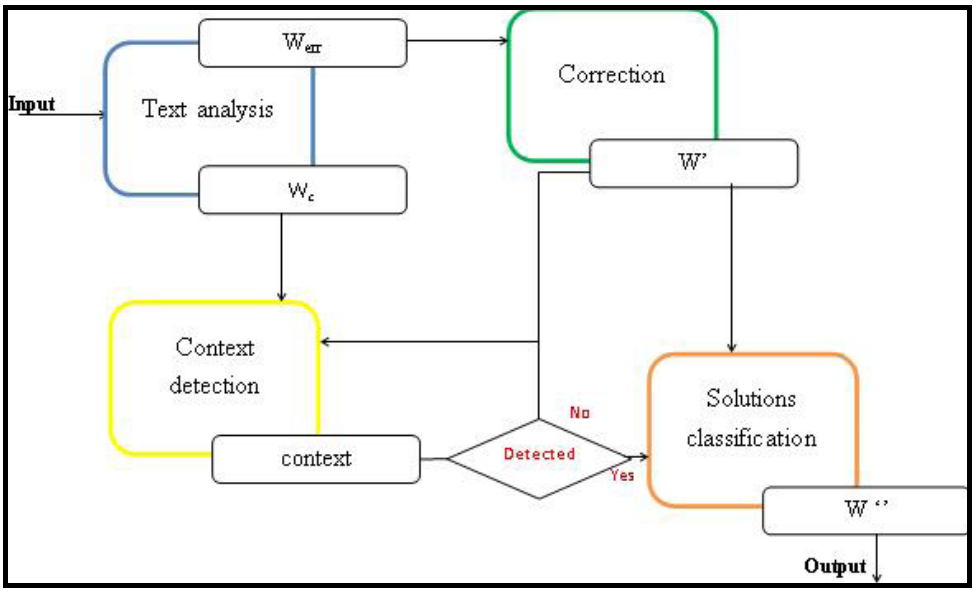
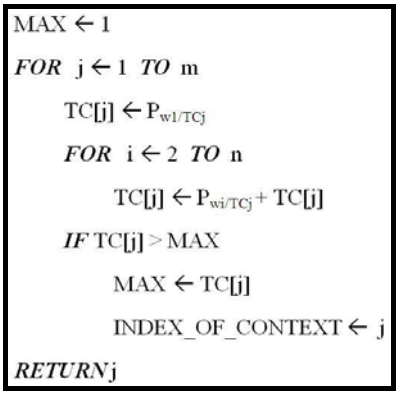
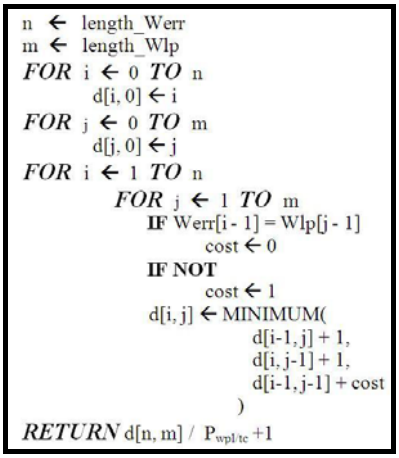


Figure 2 Algorithm of detection of context



The complexity of the algorithm is $O(m * n)$, where m is the total number of the words in the paragraph e and n is the total number of the contexts.

Figure 3 Algorithm of correction



The complexity of the algorithm is $O(m * n)$, where n and m are the length of W_{err} and W_{lp}.

5 Test and evaluation

5.1 The implementation of our approach

To assess our approach, we have developed a tool called Arabic corrector words in its context. The latter is a topical context correction system based on a database of 7,250,233 words including various types of elements, namely: names, verbs, significant terms which can identify the context, and others. In addition to this database, there exists the corpus which we created previously. This corpus provides information about the relative probability of terms carrying the information to identify the context in different contexts.

Arabic corrector words in its context have several parameterised options for several objectives among which we quote: the addition of the lexicons in the corpus of learning, the correction off context, the correction according to the context and others. Besides, the system automatically ignores the number, dates, e-mail addresses and alphanumeric strings.

5.2 Experimental results

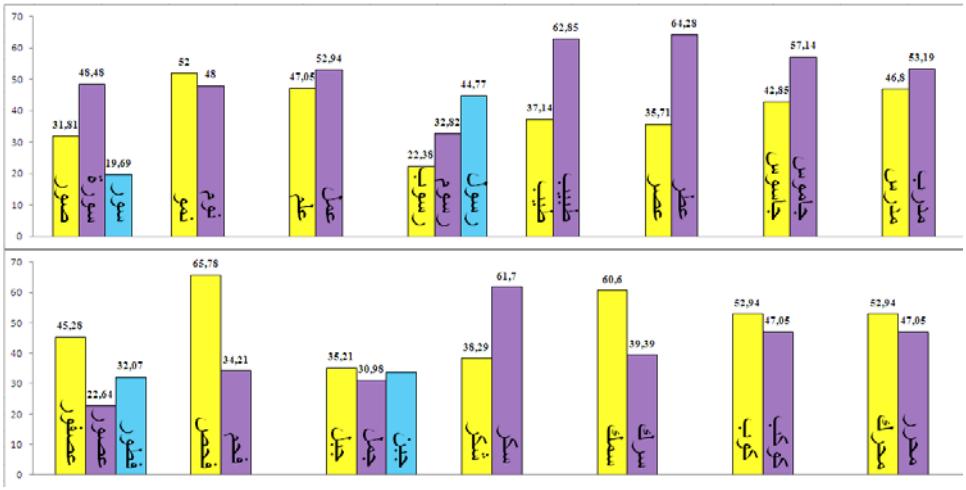
In order to evaluate our approach, we have proceeded like follows:

- the test consists on demonstrating the impact of the context in the correction by comparing the correction without context based on the Levenshtein algorithm and the correction according to the context based on our approach
- a comparison between our approach and the approach of the language model (Aouragh et al., 2015).

5.2.1 Test 1

In the following experiments, the test is done on a test corpus and 15 sets of confusion selected based on their frequency in the Arabic language. For the test corpus, we have considered a set of paragraphs in ten contexts having each words taking part in the confusion set. The illustration bellow demonstrates the appearance rate of every word in the text according to its confusion set. For example, for the confusion set [سورة, سور صور], the word صور appears 21 times in the test corpus (31.81%), the word سورة appears 32 times (48.48%) and the word سور appears 13 times (19.69%).

Figure 4 The apparition rate of each word in the text (see online version for colours)



In each paragraph, we have produced errors on the words that belong to the confusion set. The test aims to detect the issue in the first place, and then remove the confusion produced by combining the words while putting in the first place the most suitable solution for the current context. We have divided the test in two parts: the correction off

context and the correction according to the context. The test consists in comparing the appearance frequency of each word in the confusion sets according to its initial state. The results are expressed by the rate of apparition of the word as a solution on each confusion set.

Based on this analysis, we notice that the context has a positive impact on the correction. It allows sorting and ordering the solutions according to the text's content while putting in the first place the most appropriate solution for the context. For example, in the correction out context, we notice that the appearance frequency of the words سورة of the first confusion set has decreased. This cutting is compensated by a raise of the word سور and the word صور which means that the words سورة has not been suggested as first solutions for the erroneous words, when the correction in context, the appearance frequency of words of the same confusion set remains the same as for the initial state.

Figure 5 Results of the test off context (see online version for colours)

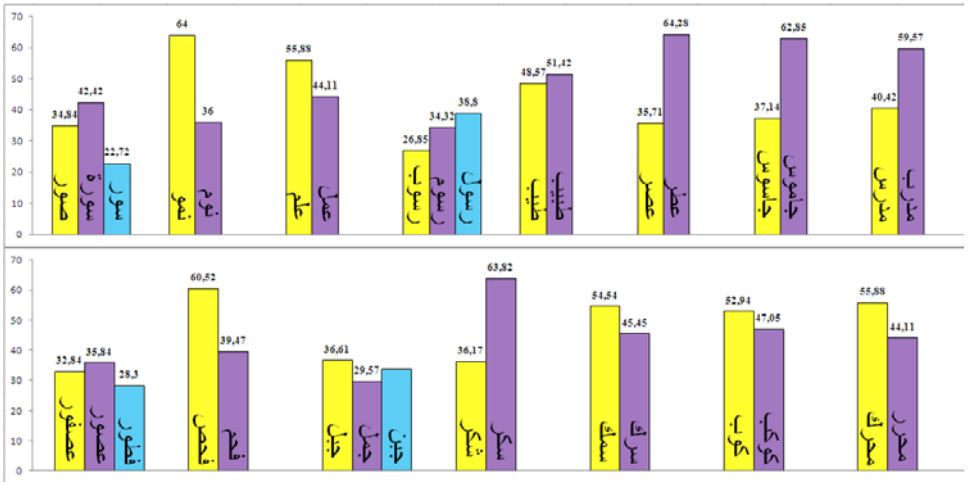
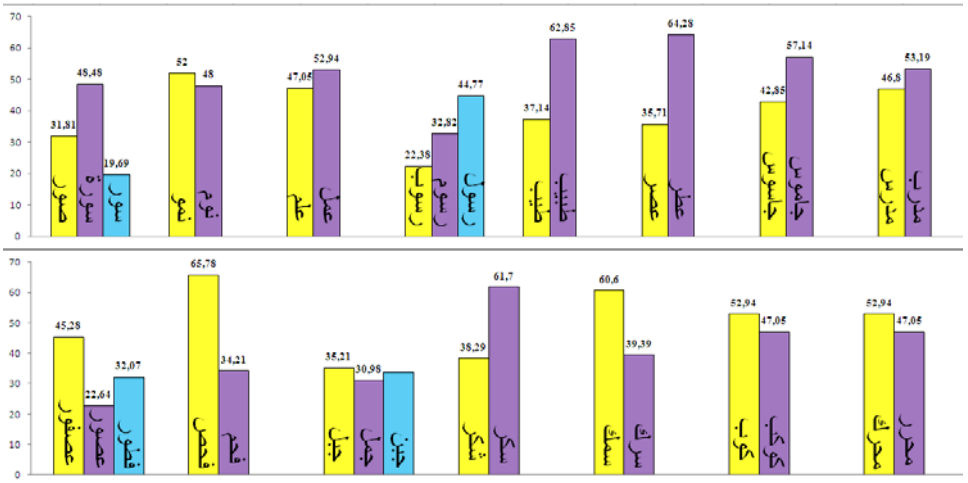


Figure 6 Results of the test according to the context (see online version for colours)



5.2.2 Test 2

In the following experiments, we have compared our method with the algorithm of Aouragh et al. (2015). We chose this algorithm because of it shares some points with our algorithm that is the correction of the Arabic language and the detection of errors thanks to Levenstein. Experiments are conducted on a corpus of text in Arabic. To evaluate the Aourache technique, we used a corpus of learning that we have cut into segments of bigram by a program that we have developed. This program allows to extract the bigram from the documents that we used to create the corpus of Arabic corrector words in its context learning.

Once an error is detected, Arabic corrector words in its context offers a set of solutions arranged in the context of the words surrounding the error by putting in the first place the one having a higher probability detected in the context. This probability is estimated in its learning corpus. The bigram technique offers a set of solutions like Arabic corrector words in its context and it classifies these solutions according to the probability of occurrence of the solution knowing that the words surrounding the error appeared.

The purpose of the test is to detect the error in the first place, and then remove the confusion produced by linking the words while putting in first position the most suitable solution for the current context. The results are expressed in percentage of correct classifications on each set of confusion.

Once the tests have been carried out, we have achieved the following results set:

Table 1 Comparative table between our method and method of the model bi-grams

	Method by using model bi-grams	*Our solution*
First position	78%	93%
Second position	6%	5%
Third position	14%	2%
Other position	2%	0%
Execution time (second)	5,261.04	2,752.23

Source: Aouragh et al. (2015)

According to these results, we notice that the correction rate of our solution is higher compared to algorithms based on n-gram model. This could be explained by the fact that the n-gram model is based on an n size history to build a probability distribution for the next word which means that the word prediction depends on the order of the set of the $n - 1$ preceding words, while the proposed solution is based on the key words carrying the information and composing the paragraph. These words are used to identify the context that will help to predict the next word whichever the preceding word's order is. To illustrate that case, we consider the following sentence.

دخل المعلم الى القسم لإجراء التمارين بعد إنهاء الدرس بحضور جميع التلاميذ (daxala AlmuEalimu <laY Alqqm l<ijora' AltamaAriyn baEoda <inohaA' Aldarosi biHuduwr jamiyE AltalAmiy* – The teacher went to the class to do exercises after finishing the lesson in the presence of all the students).

Figure 7 Correction sample off context (see online version for colours)

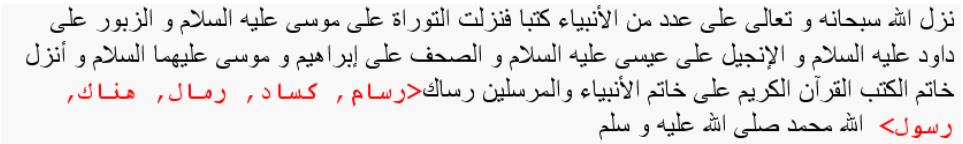
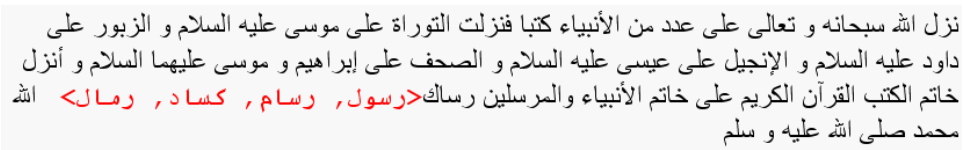


Figure 8 Correction sample according to the context (see online version for colours)



The word الفقم (Alqqm) is a misspelled word. Among the proposed solutions, we find القسم (Alqisomi – the class) and القدم (Alqadam – the foot). To choose the most suitable solution, we try to identify the sentence's context according to the following words المعلم (AlmuEalimu – the teacher), التمارين (AltamaAriyn – exercices), الدرس (Aldarosi – the lesson) and التلاميذ (AltalAmiy* – the students). According to our knowledge corpus, these words have a higher probability in the education field than in others. We conclude then that the context of the sentence is education, which drives us to put the word القسم (Alqisomi – class) in the first place as the most suitable solution for the context. However, in the n-gram model, it is difficult to identify which solution [القسم (Alqisomi – class) and القدم (Alqadam – the foot)] is the most suitable for the context. This is due to the lexicon insufficiency that is expressed by:

- the absence of the set of $n-1$ words preceding the misspelled one in the to knowledge corpus (its probability is then fixed to 0)
- the probability (W'_n / W_{n-1}) (W'_n no suitable solution according to the context) is higher than the probability (W_n/W_{n-1}) (W_n the most suitable solution according to the context), because the estimation of these probabilities is not always possible for the Arabic language due to it is morphological and semantic richness.

6 Conclusions

In this paper we proposed an automatic spelling correction method according to the context, the aim is to increase the accuracy of the solutions proposed by an automatic spell checker. Our approach uses a rich vocabulary and relies on the combination of both the lexical correction and the correction according to the context using a well shaped corpus. Regarding the lexical level, we proposed for a wrong word the closest lexical words based on a typographical correction rules (deletion, addition of a character, substitution of a letter with another). We use the Levenshtein algorithm to compute the minimum distance between the lexical words and the wrong word. Concerning the topical context level, we used a learning corpus containing a probability distribution of the appearance of a word in different context. The latter is built by bringing a collection of documents available in the internet. To implement our idea, we compared it with n-gram

algorithm. The obtained results show that the contribution of the information in the context eliminates more candidates. Our new approach brings several advantages. These advantages concern several axes in the automatic spelling correction procedure, the most important are the reduction of number of parameters to be estimated and the execution time which became more and more reduced.

References

Abdelhadi, S., Bosch, V. and Günter, A. (2007) 'Arabic computational morphology, knowledge-based and empirical methods', *Speech and Language Technology*, Vol. 38, pp.206–217.

Aho, A.V. and Corasick, M.J. (1975) 'Efficient string matching: an aid to bibliographic search', *Communications of the ACM*, Vol. 18, No. 6, pp.333–340.

Aouragh, L., Yousfi, A. and Gueddah, H. (2015) 'Adaptating the Levenshtein distance to contextual spelling, correction', *International Journal of Computer Science and Applications*, Vol. 12, No. 1, pp.127–133.

Attia, M., Pecina, P., Samih, Y., Shaalan, K. and Genabith, V.J. (2015) 'Arabic spelling error detection and correction', *Natural Language Engineering*, Vol. 22, No. 3, pp.1–23.

Bacchiani, M., Hirschberg, J., Rosenberg, A., Whittaker, S., Hindle, D., Isenhour, P., Jones, M., Stark, L. and Zamchick, G. (2001) 'Audio navigation in the voicemail domain', *Human Language Technology Conference*.

Bakkali, H., Yousfi, A., Gueddah, H. and Belkasmi, M. (2014) 'For an independent spell-checking system from the Arabic language vocabulary', *International Journal of Advanced Computer Science and Applications*, Vol. 5, No. 1, pp.113–116.

Ben Othmane, Z. and Ben Ahmed, M. (2003) 'Le contexte au service de la correction des graphies fautives arabes', *conférence sur le Traitement Automatique des Langues Naturelles TALN 2003*, pp.11–14.

Brown, P., Cocke, J., Della, P.S., Della, P.V., Jelinek, F., Lafferty, J., Mercer, R. and Roossin, P. (1990) 'A statistical approach to machine translation', *Computational Linguistics*, Vol. 16, No. 2, pp.79–85.

Cattoni, R., Federico, M. and Lavie, A. (2001) 'Robust analysis of spoken input combining statistical and knowledge-based information sources', *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop*, pp.347–350.

Chadben, N. and Belguith, H.L. (2003) 'L'dtiquetage morpho-syntaxique: comment lever l'ambiguite dans les textes arabes non voyellds?', *troisieme journees scientifiques des jeunes chercheurs en genie eiectrique et informatiquei*, pp.41–44.

Damerau, F. (1964) 'A technique for computer detection and correction of spelling errors', *Communications of the ACM*, Vol. 7, No. 3, pp.659–664.

Debili, F., Achour, H. and Souissi, E. (2002) 'La langue arabe et l'ordinateur, de l'étiquetage grammatical à la voyellation automatique', *IRMC*, No. 71.

Fossati, D. and Di Eugenio, B. (2007) 'A mixed trigrams approach for context sensitive spell checking', *International Conference on Intelligent Text Processing and Computational Linguistics*, pp.623–633.

Gauvain, J.L., Lamel, L. and Adda, G. (2000) 'Transcribing broadcast news for audio and video indexing', *Communications of the ACM*, Vol. 43, No. 2, pp.64–70.

Golding, A. and Roth, D. (1999) 'A winnow-based approach to context-sensitive spelling correction', *Journal Machine Learning – Special Issue on Natural Language Learning*, Vol. 34, Nos. 1–3, pp.107–130.

Gueddah, H., Yousfi, A. and Belkasmi, M. (2012) 'Introduction of the weight edition errors in the Levenshtein distance', *International Journal of Advanced Research in Artificial Intelligence*, Vol. 1, No. 5, pp.30–32.

Hassan, Y., Aly, M. and Atiya, A. (2014) 'Arabic spelling correction using supervised learning', *Proceeding of the EMNLP 2014 Workshop on Arabic Natural Language Processing*, pp.121–126.

Jelinek, F. (2001) 'Aspects of the statistical approach to speech recognition', *IEEE International Symposium on Information Theory*.

Johnson, S., Jourlin, P., Moore, G., Sparck, J.K. and Woodland, P. (1999) 'The Cambridge University spoken document retrieval', *ICASSP 99*, No. 2304.

Knuth, D. (1973) *The Art of Computer Programming – Sorting and Searching Volume 3*, Addison-Wesley Publishing Company Reading Massachusetts, Vol. 3.

Kukich, K. (1992) 'Techniques for automatically correcting words in text', *ACM Computing Surveys*, Vol. 24, No. 4, pp.39–477.

Levenshtein, V. (1966) 'Binary codes capable of correcting deletions, insertions and reversals', *SOL Phys. Dokl*, Vol. 10, No. 8, pp.707–710.

Mitton, R. (1996) *English Spelling and the Computer*, Longman Group, Harlow Essex.

Mustafa, M., AbdAlla, H. and Suleman, H. (2008) 'Current approaches in Arabic IR: a survey', *Proceedings the Annual International Conference on Asia-Pacific Digital Libraries (ICADL)*, Vol. 5362, pp.406–407.

Nejja, M. and Yousfi, A. (2015) 'A lightweight system for correction of Arabic derived words', *Mediterranean Conference on Information and Communication Technologies*, Vol. 380, pp.131–138.

Nwesri, F.A., Tahaghoghi, S.M.M. and Scholer, F. (2005) 'Stemming Arabic conjunctions and prepositions', *Proceedings of the International Symposium on String Processing and Information Retrieval (SPIRE)*, Vol. 3772, pp.206–217.

Oflazer, K. (1996) 'Error-tolerant finite-state recognition with applications to morphological analysis and spelling correction', *Computational Linguistics Archive*, Vol. 22, No. 2, pp.73–89.

Peterson, J.L. (1980) 'Computer programs for detecting and correcting spelling errors', *Comm. ACM*, Vol. 23, No. 12, pp.676–687.

Pollock, J.J. and Zamora, A. (1984) 'Automatic spelling correction in scientific and scholarly text, *Communications of the ACM*, Vol. 27, No. 4, pp.358–368.

Rosenfeld, R. (2000) 'Two decades of statistical language modeling: where do we go from here?', *Proceedings of the IEEE*, Vol. 88, No. 8, pp.1270–1278.

Savary, A. (2000) *Recensement et description des mots composés – méthodes et applications*, Thèse de doctorat en Informatique Fondamentale, Université de Marne-la-Vallée, pp.149–158.

Shaalan, K., Allam, A. and Gomah, A. (2003) 'Towards automatic spell checking for Arabic', *Proceedings of the 4th Conference on Language Engineering, Egyptian Society of Language Engineering*, pp.240–247.

Shaalan, K., Samih, Y., Attia, M., Pecina, P. and Genabith, V.J. (2012) 'Arabic word generation and modelling for spell checking', *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pp.719–725.

Srihari, R. and Baltus, C. (1992) 'Combining statistical and syntactic methods in recognizing handwritten sentences', *AAAI Symposium: Probabilistic Approaches to Natural Language*, pp.121–127.

Ukkonen, U. (1992) 'Approximate string matching with q-grams and maximal matches', *Theoretical Computer Science*, Vol. 92, No. 1, pp.191–211.

Wehr, H. (1961) *Dictionary of Modern Written Arabic*, Otto Harrassowitz, Weisbaden, Germany.

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', *Proceedings of CICLing-2008*, Vol. 4919, pp.605–616.

Woszczyna, M. (1998) *Fast Speaker Independant Large Vocabulary Continuous Speech Recognition*, These de Punirersite de Karlsruhe, Allemagne, p.156.

Xu, W., Tetteault, J., Chodorow, M., Grishman, R. and Zhao, L. (2011) 'Exploiting syntactic and distributional information for spelling correction with web-scale N-gram models', *Proceedings of Conference on Empirical Methods in Natural Language Processing*, pp.1291–1300.