Contextual Text Categorization: An Improved Stemming Algorithm to Increase the Quality of Categorization in Arabic Text

Said Gadri and Abdelouahab Moussaoui
Department of Computer Science, University Ferhat Abbas of Setif, Algeria

Abstract: *One of the methods used to reduce the size of terms vocabulary in Arabic text categorization is to replace the different variants (forms) of words by their common root. This process is called stemming based on the extraction of the root. Therefore, the search of the root in Arabic or Arabic word root extraction is more difficult than in other languages since the Arabic language has a very different and difficult structure, that is because it is a very rich language with complex morphology. Many algorithms are proposed in this field. Some of them are based on morphological rules and grammatical patterns, thus they are quite difficult and require deep linguistic knowledge. Others are statistical, so they are less difficult and based only on some calculations. In this paper we propose an improved stemming algorithm based on the extraction of the root and the technique of n-grams which permit to return Arabic words' stems without using any morphological rules or grammatical patterns.*

Keywords: *Root extraction, information retrieval, bigrams, stemming, Arabic morphological rules, feature selection.*

Received February 22, 2015; accepted August 12, 2015

1. Introduction

Arabic is one of the oldest and the most used language in the world. It is spoken by over 300 million people in the Arabic world, and used by more than 1.7 billion Muslims over the world because it is the language of the Holy Quran. Here we can distinguish two types of Arabic; a more classical language, as found in the Holy Quran or poetry, a standardized modern language, and regional dialects [3]. We note also that the Arabic language is a Semitic language [13, 16] based on 28 cursives letters written from right to left. The word in Arabic is formed of the root part and some affixes (antefixes, prefixes, infixes, suffixes) that form the word (سألتمونيها Saaltmwnyha). The Arabic root extraction is a very difficult task. This is not the case for other languages as English or French, because Arabic is a very rich language with a very difficult structure and complex morphology. Arabian linguists show that all nouns and verbs of the Arabic language are derived from a set of roots containing about 11347 roots; more than 75 % of them are trilateral roots [4]. There are many applications based on the roots of words in Arabic processing such as: text's classification, text summarizing, information retrieval, data and text mining [18, 29]. The Arabic word's roots can be classified according to the vowel letters (ي ، و ، أ a, w, y) into two types [15]. The first one is called the strong roots that do not contain any vowel (ذهب، خرج، فتح go, come out, open), the second one is called vocalic roots that contain at least one vowel (أوى ، وعد shelter, promise). Arabic roots can be further classified according to the number of their characters into four types: Trilateral roots which form most words in the Arabic language [4] (e.g., علم ، كتب ، خرج know, write, come out), Quadrilateral roots(e.g., دحرج ، طمأن roll, assure), Quinquelateral roots (e.g., انكسر، اقتصد ، انطلق ‹broken, economize, start) and Hexalateral roots (استعمل ، استحسن اقشعّر use, enjoy, tremble). There are two classes of methods used to extract the roots of Arabic words. The first class is based on morphological rules. So its methods simulate the same process as that of an expert linguist during his analysis of a given Arabic word [3, 7, 8, 23, 27], which makes the process of extracting a root difficult and complex because of the diversity of morphological formulas and the multiplicity of word forms for the same root when changing the original characters position in the word (e.g., علم ، عالم ، علوم ، عوالم ، معالم know, scientist, sciences, worlds, landmarks) [6, 19]. The second class is formed of statistical methods which are simple, fast, and do not require any morphological rules but some calculations [1, 5, 9, 10, 12, 17, 21].

In this paper, we propose an improved statistical algorithm which permits to build an Arabic stemmer based on the extraction of words' roots and the approach of n-grams of characters without using any morphological rule. The paper is organized as follows: the first section presents some related works, so we review some papers that treat the problem of extraction of Arabic words' roots. In the second section we introduce our new algorithm. The third section presents the experiments that we have done to test our new method and also displays the obtained results. In the last section we conclude our work with a summary and some ideas to improve it in the future.

2. Related Work

Many researchers proposed some algorithms to extract Arabic words roots, some of these algorithms are based on morphological rules. Thus, they are called morphological methods. Others do not use any morphological rule but some statistical calculations, so they are called statistical algorithms. In the first class of algorithms, we can note the following:

Khoja's roots [22, 23] extractor removes the longest suffix and the longest prefix. It then, matches the remaining word with verbal and noun patterns, to extract the root. The roots extractor makes use of several linguistic data files such as a list of all diacritics, punctuation characters, definite articles, and stop words [24, 25, 26, 28].

Reference [6] proposed a linguistic approach for root extraction as a pre-processing step for Arabic text mining. The proposed approach is composed of a rule-based light stemmer and a pattern-based infix remover. They propose an algorithm to handle weak, eliminated-long vowel, hamzated and geminated words. The accuracy of the extracted roots is determined by comparing them with a predefined list of 5,405 trilateral and quadrilateral roots. The linguistic approach performance was tested on in-house texts collection consisting in eight categories, the author achieved a success ratio about 73.74%.

Reference [2] presented a new Arabic root extraction algorithm that tries to assign a unique root for each Arabic word without having an Arabic roots list, a words patterns list, or the Arabic words' prefixes and suffixes list. The algorithm predicts the letter positions that may form the word root one by one, using rules based on the relations between the Arabic word letters and their placement in the word. This algorithm consists in two parts. The first part gives the rules that distinguish between the Arabic definite letter "الـ AL, La" and the original word letters "الـ". The second part segments each word into three parts and classifies its letters according to their positions. The author tested her proposed algorithm using the Holy Quran words and obtained an accuracy of 93.7% in the root extracting process.

In the second class of algorithms, we can note the following:

Reference [9] developed a root extraction algorithm which does not use any dictionary. Their algorithm categorizes all Arabic letters according to six integer weights, ranging from 0 to 5, as well as the rank of the letter which is determined by the position this letter holds in a word. The weight and rank are multiplied together, and the three letters with the smallest product constitute the root of the word. We note that [9] did not explain or clarify why or on which basis did he use such ranking or weighting.

Reference [27] proposed an algorithm to extract tri-literal Arabic roots, this algorithm consists in two steps; in the first step, they eliminate stop words as well prefixes and suffixes. In the next step, they remove the repeated words' letters until only three letters are remain. Then they arrange these remaining letters according to their order in the original word, which form the root of the original word. The obtained results of this algorithm were very promising and give an accuracy of root's extraction over 73%.

Reference [20] proposed a new way to extract the roots of Arabic words using n-grams technique. They used two similarity measures; the dissimilarity measurement, or the "Manhattan distance measurement" and the "Dice's measurement". They tested their algorithm on the Holy Quran and on a corpus of 242 abstracts from the Proceedings of the Saudi Arabian National Computer Conferences. They concluded from their study that combining the n-grams with the Dice's measurement gives better results than using the Manhattan distance measurement.

Reference [11] proposed a new algorithm to find the system that assigns, for every non vowel word a unique root depending on the context of the word in the sentence. The proposed system consists in two modules; the first one consists in analysing the context by segmenting the words of the sentence into its elementary morphological units in order to extract its possible roots. So, each word is segmented into three parts (prefix, stem and suffix). In the second module, they based on the context to extract the correct root among all possible roots of the word. For this purpose, they used a Hidden Markov Models (HMM) approach, where the observations are the words and the possible roots are the hidden states. They validate their algorithm using NEMLAR Arabic writing corpus that consists in 500,000 words, and their proposed algorithm gives the correct root in more than 98% of the training set and 94% of the testing set.

Reference [30] proposed a new algorithm which uses the n-grams technique. An n-gram is a basic text analysis tool that is used in natural language processing. In this technique, both the word and its assumed root are divided into pairs (called bi-gram, or di-gram) then the similarity between the word and the root is calculated using Equation (1) [14]. This process is repeated for each root in the roots list:

$$S = 2 * C / (A + B) \qquad (1)$$

Where:
A = Number of unique bi-grams in the word (A)
B = Number of unique bi-grams in the root (B)
C = Number of similar unique pairs between the word (A) and the root (B)

To use Equation (1) for extracting the word's root, we must have: the word (*A*) and the potential roots (*B*) to compare with, then the similarity measuring is conducted by computing the value of (*S*) between the word (*A*) and each potential roots (*B*).

3. The Proposed Algorithm

In our new algorithm, we also use the n-grams technique to extract Arabic words roots. For this purpose, we proceed according to the following steps:

- *Step 1*. We segment the word for which we want to find the root, and all the roots of the list into bigrams (2-grams).

For example if we have the word “يذهبون” and a list of six (06) roots (فتح ، خرج ، ذهب ، وجد ، وهب ، نهب), we proceed the segmentation step as follows:
W = “يذهبون” ➔ (يذ، يه ، يب ، يو ، ين ، ذه ، ذب ، ذو ، ذن ، هب
، هو ، هن ، بو ، بن ، ون)
R_1 = “فتح” ➔ (فح ، تح ، فت)
R_2 = “خرج” ➔ (خر ، خج ، رج)
R_3 = “ذهب” ➔ (ذه ، ذب ، هب)
R_4 = “وجد” ➔ (وج ، ود ، جد)
R_5 = “وهب” ➔ (وه ، وب ، هب)
R_6 = “نهب” ➔ (نه ، نب ، هب)

- *Step 2*. We calculate the following parameters:
  N_w : The number of bigrams in the word *w*
  N_{R_i} : The number of bigrams in the root *Ri*
  N_{wR_i} : The number of common bigrams between the word *W* and the root *Ri*
  $N_{w\overline{R_i}}$: The number of bigrams belonging to the word w and do not belong to the root *Ri* ($N_{w\overline{R_i}} = N_w$ - N_{wR_i})
  $N_{R_i\overline{w}}$: The number of bigrams belonging to the root *Ri* and do not belong to the word *w* ($N_{R_i\overline{w}} = N_{R_i} - N_{wR_i}$) .

For the previous example we have:

$$N_W = 15, N_{R_1} = 3, N_{R_2} = 3, N_{R_3} = 3, N_{R_4} = 3, N_{R_5} = 3,$$
$$N_{R_6} = 3, N_{WR_1} = 0, N_{WR_2} = 0, N_{WR_3} = 3, N_{WR_4} = 0, N_{WR_5} = 1,$$
$$N_{WR_6} = 1, N_{W\overline{R_1}} = 15, N_{W\overline{R_2}} = 15, N_{W\overline{R_3}} = 12, N_{W\overline{R_4}} = 15,$$
$$N_{W\overline{R_5}} = 14, N_{W\overline{R_6}} = 14, N_{R_1\overline{W}} = 3, N_{R_2\overline{W}} = 3, N_{R_3\overline{W}} = 0,$$
$$N_{R_4\overline{W}} = 3, N_{R_5\overline{W}} = 2, N_{R_6\overline{W}} = 2$$

- *Step 3*. We take only the roots having at least one common bigram with the word *w* ($N_{wR_i} \geq$ 1) as candidate roots among the list of all roots in order to reduce the calculation time.

In our previous example, we can take only the roots: R_3 = “ذهب”, R_5 = “وهب”, R_6 = “نهب” with N_{wR_i} = 3, 1, 1 respectively.

- *Step 4*. we calculate the distance *D(w, R_i)* between the word *W* and each candidate root R_i (R_3, R_5, R_6) according to the following equation :

$$D(w, R_i) = 2 * N_{wR_i} + K * N_{w\overline{R_i}} + k * N_{R_i\overline{w}} \qquad (2)$$

Where: *k* is a constant which must take a high value (we put here *k*=100)
For the previous example we obtain:
*D(w, R_3) = 2*3+100*12+100*0 = 1206*
*D(w, R_5) = 2*1+100*14+100*2 = 1602*
*D(w, R_6) = 2*1+100*14+100*2 = 1602*

- *Step 5*. In the last step, we assign the root that has the lowest value of distance *D(w, R_i)* among the candidate roots to the word *W*. It is the required root.

In our example, the root of the word “يذهبون” is “ذهب”
Finally, we note that our new algorithm has the following advantages:

1. Does not require the removal of affixes whose distinction from the native letters of the word is quite difficult.
2. Works for any word whatever the length of the root, i.e., 3-lateral, 4-drilateral, 5-lateral, 6-lateral roots.
3. Valid for strong and vocalic roots which generally pose problems in Arabic during their derivation, because of the complete change of their forms.
4. Does not use any morphological rule nor patterns but only simple calculations of distances.
5. Very practical algorithm and easy to implement on machine.

4. Experimentations and Obtained Results

To validate our proposed algorithm, we used three corpuses which can be classified according their sizes into: small corpus, middle corpus, and large corpus (see Table 1). Each one is constituted of many files as indicated below:

1. The file of derived forms (gross words) which contains morphological forms of words derived from many Arabic roots.
2. The file of roots which contains many Arabic roots. We note that these roots are trilateral, quadrilateral, quinquelateral, and hexalateral. We also note that many of them are vocalic roots which contain at least one vowel (see Table 2).
3. The file of golden roots which contains the correct roots of all words present in our corpus. This golden list was prepared by an expert linguist and used as a reference list, i.e., by comparing the list of obtained roots (extracted by the system) and the reference list (established by the expert), we can calculate the roots extraction accuracy (success ratio). The extraction process and the obtained

results are shown in Tables 3 and 4 as well as Figures 1 and 2.

Table 1. Corpuses used in experiments.

Corpus	Size of derived words' file	Size of the roots' file	Size of the golden roots' file
Small corpus	50	25	50
Middle corpus	270	135	270
Large corpus	2250	600	2250

Table 2. A sample of 3-lateral, 4-lateral, 5-lateral, 6-lateral roots.

Trilateral roots	Quadrilateral roots	Quinquelateral roots	Hexalateral roots
زرع	أكرم	انطلق	استعمل
صنع	أعان	انكسر	استحسن
تجر	أعطى	احتوى	استعان
أبى	علّم	اجتمع	اعشوشب
نفر	ربّى	اخضرّ	ادهامّ
على	برّأ	تقدّم	اخضارّ
دار	قاتل	تحطّم	اجلودّ
طار	حاسب	تحدّى	احرنجم
عطس	داين	تعاطى	افرنقع
صدع	زلزل	تدحرج	اطمأنّ

Table 3. Extraction of some Arabic roots using our new algorithm.

Word	Nearest roots	Nb.Common bi-grams	Distance values	Extracted root	Correct root
يتعلمون	علم ، عالج ، كلّم ، عمل ، كمن ،	3 ، 1 ، 3 ، 2 ، 1	2806 ، 3202 ، **2506** ، 2704 ، 2902،	علم	علم
كاتب	اقتصد ، كتب	1 ، 3	1402 ، **306**	كتب	كتب
معلم	علم ، عالج ، كلّم ، عمل ، كمل ،	3 ، 1 ، 3 ، 3 ، 1	1006 ، 1402 ، **706** ، 708 ، 1102	علم	علم
كتاتيب	اقتصد، كتب، تأتأ	1 ، 3 ، 1	2002 ، **906** ، 1402	كتب	كتب
اقتصاد	قصد ، اقتصد ، عقد	3 ، 10 ، 1	1106 ، **420** ، 1502	اقتصد	اقتصد
يقصدون	قصد ، اقتصد ، عقد	3 ، 3 ، 1	**1206** ، 1906 ، 1602	قصد	قصد
علاج	عالج ، علم ، عمل	5 ، 1 ، 1	**210** ، 702 ، 702	عالج	عالج
معالجة	عالج ، علم ، عمل ، كمل	6 ، 1 ، 2 ، 1	**912** ، 1602 ، 1404 ، 1602	عالج	عالج
استخدم	اقتصد ، خمد ، خدم	3 ، 2 ، 3	1906 ، 1404 ، **1206**	خدم	خدم
خادم	اقتصد ، خمد ، خدم	1 ، 2 ، 3	1402 ، 504 ، **306**	خدم	خدم
كمون	كلم ، كمل، كمن	1 ، 1 ، 3	702 ، 702 ، **306**	كمن	كمن
سنستدرجهم	اقتصد ، خدم ، درج ، هزم	1 ، 1 ، 3 ، 1	3802 ، 3102 ، **2706** ، 3102	درج	درج
متذبذب	كتب ، ذبذب	1 ، 4	1002 ، **508**	ذبذب	ذبذب
هزائم	هزم	3	**706**	هزم	هزم
يهزمونهم	كمن ، هزم	1 ، 3	2006 ، **2402**	هزم	هزم
المربّون	كلم ، عالج ، علم ، كمن ، ربّى ، طار	1 ، 1 ، 1 ، 1 ، 3 ، 1	2902 ، 3202 ، 2902 ، 2902 ، **2806** ، 2902	ربّى	ربّى
طيران	طار	2	**904**	طار	طار
طائرات	اقتصد ، طار	3 ، 1	2102, **1006**	طار	طار

Table 4. Obtained results when extracting the words roots.

Corpus	Nb.Roots	Nb.Words	Correct Results	Wrong Results	Success Rate(%)	Error Rate (%)
Small	25	50	49	1	98,00	2,00
Middle	135	270	253	17	94,07	5,93
Large	600	2250	2028	222	90,13	9,87

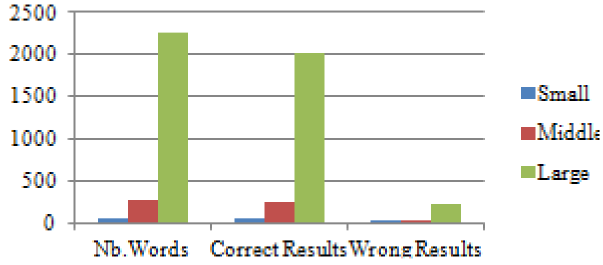

Figure 1. Correct and wrong results in number of words.

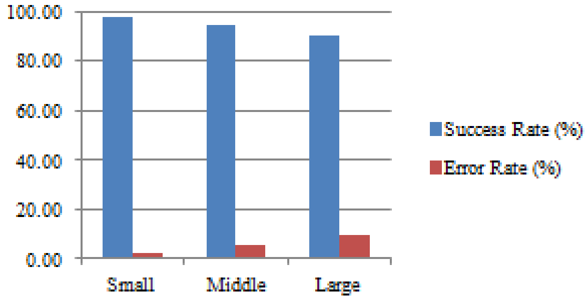

Figure 2. Calculation of success rate and error rate.

5. Comparisons with Other Algorithms

To show the effectiveness of our algorithm proposed, we concluded our work by establishing a comparison with other known algorithms. For this purpose, we took a sample words list and tried to extract the root of each word using three well-known algorithms which are: Khodja stemmer, Yousef *et al*. stemmer, and our proposed stemmer, the obtained results were shown in Table 5. On the other hand, we illustrated the obtained results when applying the three above algorithms on the three corpuses used in the experimentation, namely: the small corpus, the middle corpus, and the large corpus, then, we summarized the obtained accuracy for each algorithm in Table 6 and Figure 3.

Table 5. Extraction of some roots using the three algorithms.

Word	Extracted Root			
	Khodja Algo	Yousef Algo	Our new Algo	Corr. root
يتعلّمون	علم	علم	علم	علم
كاتب	كتب	كتب	كتب	كتب
كتاتيب	Not stemmed	كتب	كتب	كتب
اقتصاد	قصد	اقتصد	اقتصد	اقتصد
سنستدرجهم	Not stemmed	درج	درج	درج
متلألئ	Not stemmed	لألأ	لألأ	لألأ
المربّون	ربن	ربّى	ربّى	ربّى
طائرات	طور	طار	طار	طار
ولولة	ليل	ولول	ولول	ولول
وقيعة	قوع	وقع	وقع	وقع
يزنونهم	زنن	نهب	وزن	وزن
زلازل	Not stemmed	تنازل	زلزل	زلزل
حواسيب	Not stemmed	نسي	حسب	حسب
ناسج	نسج	سجد	نسج	نسج
نوازل	نزل	تنازل	نزل	نزل

Table 6. Illustration of obtained accuracy for the three algorithms.

Corpus	Size		The obtained accuracy (suc_ rate, err_ rate)%					
	Nb.roots	Nb.words	Khodja Algo		Yousef Algo		Our new Algo	
Small	25	50	68,00	32,00	92,00	8,00	98,00	2,00
Middle	135	270	83,70	16,30	63,33	36,66	94,07	5,93
Large	600	2250	68,43	31,57	61,32	38,68	90,13	9,87

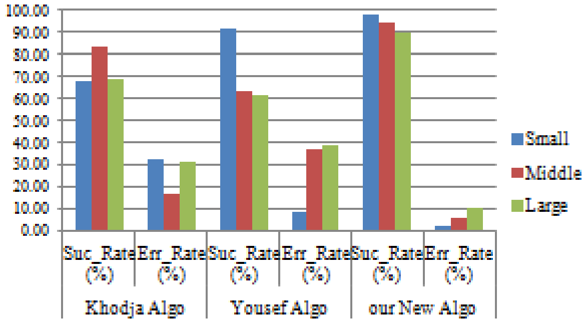

Figure 3. Comparison between three algorithms.

6. Discussion

From Table 5, we see that Khodja algorithm fails sometimes in getting the correct root of the given word and for many words it produced one of two results:

1. Not stemmed (i.e., سنستدرجهم, حواسيب).
2. A new word and sometimes a wrong word that does not exist in Arabic (i.e., (قوع ، وقيعة) (طور ، طائرات)).

The same thing can be said for Yousef *et al*. algorithm. Although it gives better results than Khodja algorithm, it fails for many words like: ((سجد، ناسج) ,(زنن، يزنونهم)). For the same cases, our algorithm always gives the correct root and the failure is very limited. From Table 6 and Figure 3, we can deduce that our proposed algorithm gives the best results for the three used corpuses with a very high accuracy. We note here the value 98 % for the small corpus, 94,07 % for the middle corpus, and 90,13 % for the large corpus.

7. Conclusions and Perspectives

In this paper we have studied how we can reduce the size of terms in Arabic text categorization by stemming. For this purpose, we exposed the most known algorithms in the field, including morphological algorithms mainly based on the use of morphological rules of Arabic, and statistical algorithms which are the newest in the field, and require only simple calculations of distances. We also proposed a new statistical algorithm based on bigrams technique. This algorithm is fast, does not require the removal of affixes nor the use of any morphological rules, capable to find all types of roots, i.e., 3-lateral, 4-lateral, 5-lateral, and 6-lateral roots. There is no difference between strong roots and vocalic roots in our new algorithm. We also established a comparison between our proposed algorithm and two other well-known algorithms in the field, namely: Khodja algorithm, Yousef *et al*. algorithm. The first one sometimes fails in getting the correct root of the given word and for many words it produced one of two results:

1. Not stemmed word.
2. A completely new word and sometimes a wrong word that does not exist in Arabic.

The same thing can be said for the second one. Although it gives better results than the first, it fails for many words. For the same cases, our new algorithm always gives the correct root, the failure is very limited, and the obtained success ratio of root extraction is very promising. In our future work, we plan to apply our new algorithm on a corpus of Arabic words with big sizes, to improve the obtained success rate, and to apply it in extracting the root of words in other languages such as English and French.

References

[1] Ababneh M., Al-Shalabi R., Kanaan G., and Al-Nobani A., “Building an Effective Rule-Based Light Stemmer for Arabic Language to Improve Search Effectiveness,” *The International Arab Journal of Information Technology*, vol. 9, no. 4, pp. 368-372, 2012.

[2] Abu Hawas F., “Exploit Relations between the Word Letters and their Placement in the Word for Arabic Root Extraction,” *Computer Science Journal*, vol. 14, no. 2, pp. 27-431, 2013.

[3] Abu Hawas F. and Keith E., “Rule-Based Approach for Arabic Root Extraction: New Rules to Directly Extract Roots of Arabic Words,” *Journal of Computing and Information Technology*, vol. 22, no. 1, pp.57-68, 2014.

[4] AlKamar R., *Computer and Arabic language computerizing*, Dar Al Kotob Al-Ilmiya, 2006

[5] AlNashashibi M., Neagu D., and Yaghi A., “Stemming Techniques for Arabic Words: A Comparative Study,” *in Proceeding of 2nd International Conference on Computer Technology and Development*, Cairo, pp. 270-276, 2010.

[6] AlNashashibi M., Neagu D., and Yaghi A., “An Improved Root Extraction Technique for Arabic Words,” *in Proceeding of 2nd International Conference on Computer Technology and Development*, Cairo, pp.264-269, 2010.

[7] Alomari A., Abuata B., and Al-kabi M., “Building and Benchmarking New Heavy/Light Arabic Stemmer,” *in Proceeding of 4th International Conference on Information and Communication Systems*, Irbid, pp. 1-6, 2013.

[8] Alshalabi R., “Pattern-Based Stemmer for Finding Arabic Roots,” *Information Technology Journal*, vol. 4, no. 1, pp. 38-43, 2005.

[9] Al-shalabi R., Kanaan G., and Al-Serhan H., “New Approach for Extracting Arabic Roots,” *in Proceeding of the International Arab Conference on Information Technology*, Alexandria, pp. 42-59, 2003.

[10] Al-Shalabi R., Kanaan G., and Ghwanmeh S., and Nour F., “Stemmer Algorithm for Arabic Words Based on Excessive Letter Locations,” *in*

Proceeding of Innovations in Information Technologies, Dubai, pp. 456-460, 2008.

[11] Boudlal A., Belahbib R., Lakhouaja A., and Mazroui A., "A Markovian Approach for Arabic Root Extraction," *The International Arab Journal of Information Technolog*, vol. 8, no. 1, pp. 91-98, 2011.

[12] Duwairi R., "Arabic Text Categorization," *The International Arab Journal of Information Technolog*, vol. 4, no. 2, pp. 125-131, 2007.

[13] Ethnologue, http://www.ethnologue.com/statistics/size, Last Visited 2014.

[14] Frakes W., *Information Retrieval: Data Structures and Algorithms*, Prentice-Hall, 1992.

[15] Wightwick J. and Gaafar M., *Arabic Verbs and Essentials of Grammar 2E (Verbs and Essentials of Grammar Series)*, McGraw-Hill, 2007.

[16] Ghazzawi S., *The Arabic Language in the Class Room*, Georgetown University, 1992.

[17] Ghwanmeh S., Kanaan G., Al-Shalabi R., and Khanfar K., "An Algorithm for Extracting the Root of Arabic Words," in Proceeding of 5th International Business Information Management Conference, Cairo, 2005.

[18] Ghwanmeh S., Kanaan G, Al-Shalabi R., and Rabab'ah S., "Enhanced Algorithm for Extracting the Root of Arabic Words," *in Proceeding of 6th International Conference on Computer Graphics, Imaging Visualization*, Tianjin, pp. 388-391, 2009.

[19] Hajjar A., Hajjar M., Zreik K., "A System for Evaluation of Arabic Root Extraction Methods," *in Proceeding of 5th International Conference on Internet and Web Applications and Services*, Barcelona, pp. 506-512, 2010.

[20] Hmeidi I., Al-Shalabi R., Al-Taani A., Najadat H., and Al-Hazaimeh S., "A Novel Approach to the Extraction of Roots from Arabic words Using Bigrams," *Journal of American Society for Information Science and Technology*, vol. 61, no. 3, pp.583-591, 2010.

[21] Kanaan G., Al-Shalabi R., and Al-Kabi M., "New Approach for Extracting Quadrilateral Arabic Roots," *Abhath Al-Yarmouk, Basic Science and Engineering*, vol. 14, no.1, pp. 51-66, 2005.

[22] Khodja S., Av:http://zeus.cs.pacificu.edu/shereen/research.htm, Last Visited 2014.

[23] Khodja S. and Garside R., "Stemming Arabic Text," Technical Report, 1999.

[24] Larkey L. and Connell M., "Arabic Information Retrieval at UMass in TREC-10," Proc.TREC 2001, Gaithersburg: NIST, 2001.

[25] Larkey S., Ballesteros L., and Connell M., "Improving Stemming for Arabic Information Retrieval: Light Stemming and Occurrence Analysis," *in Proceeding of 25th ACM International Conference on Research and Development on IR*, Tampere, pp. 275-282, 2002.

[26] Larkey S., Ballesteros L., and Connell M., "Light Stemming for Arabic Information Retrieval: Arabic Computational Morphology Text," *Speech and Language Technology*, vol. 38, pp. 221-243, 2007.

[27] Momani M. and Faraj J., "A Novel Algorithm to Extract Tri-Literal Arabic Roots," *in Proceeding of IEEE/ACS International Conference on Computer Systems And Applications*, Amman, pp. 309-315, 2007.

[28] Sawalha M. and Atwell E., "Comparative Evaluation of Arabic Language Morphological Analyzers and Stemmers," *in Proceeding of 2nd International Conference on Computational Linguistics*, Manchester, pp. 107-110, 2008.

[29] Yousef N., Al-Bidewi I., and Fayoumi M., "Evaluation of Different Query Expansion Techniques and Using Different Similarity Measures in Arabic Documents," *European Journal of Scientific Research*, vol. 43, pp.156-166, 2010.

[30] Yousef N., Abu-Errub A., Odeh A., and Khafejeh H., "An Improved Arabic Word's Roots Extraction Method Using N-gram Technique," *Journal of Computer Science*, vol. 10, no. 4, 2014.

Said Gadri Received his degree of engineer in computer science from the University of Setif, Algeria in 1996, and the degree of magister (Bac+7) from the University of M'sila, Algeria in 2006. He has been an assistant professor at the University of M'sila, Algeria since 2007. He is a member of the scientific council of mathematics and computer sciences faculty since 2009, member of the teaching committee of the ICST department since 2013. Currently, he is interested in many areas of research such as: Text categorization, machine learning, Text mining, Information retrieval, and natural language processing. He published many papers in different international conferences around the world.

Abdelouahab Moussaoui is Professor at Ferhat Abbas University. He received his BSc in Computer Science in 1990 from the Department of Computer Science from the University of Science and Technology of Houari Boumedienne (USTHB), Algeria. He received also an MSc degree in Machine Learning from Reims University (France) since 1992 and Master's degree in Computer Science in 1995 from University of Sidi Bel-abbes, Algeria and PhD degree in Computer Science from Ferhat Abbas University, Algeria where he obtains a status of full-professor in Computer Science. He is IEEE Member and AJIT, IJMMIA & IJSC Referee. His researches are in the areas of clustering algorithms and multivariate image classification applications. His current research interests include the fuzzy neuronal network and non parametric classification using unsupervised knowledge system applied to biomedical image segmentation and bioinformatics. He also works from a long time on pattern recognition's algorithm, complex data mining and medical image analysis.