Arabic Texts Categorization: Features Selection Based on the Extraction of Words' Roots

Said Gadri, Abdelouahab Moussaoui

Arabic Texts Categorization: Features Selection Based on the Extraction of Words' Roots

Said Gadri[1,*], Abdelouahab Moussaoui[2]

[1] Department of ICST, University of M'sila, 28000, Algeria
kadri.said28@yahoo.fr
[2] Department of Computer Sciences, University Farhat Abbes of Setif, Setif, 19000, Algeria
moussaoui.abdel@gmail.com

Abstract. One of 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. The search of root in Arabic or Arabic word root extraction is more difficult than other languages since 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 a new statistical algorithm which permits to extract roots of Arabic words using the technique of n-grams of characters without using any morphological rule or grammatical patterns.

Keywords: root extraction, information retrieval, bigrams technique, Arabic morphological rules, feature selection.

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 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 [1]. We note also that Arabic language is a semitic language [2, 3] 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 which 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 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. [5,6].

The Arabic words ‘ roots can be classified according to the vowels letters (ا ، و ، ي a, w, y) into two types [7], strong roots that do not contain any vowel (ذهب ، خرج ، فتح go, come out, open), 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 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 of an expert linguist during his analysis of a given Arabic word [1], [8,9,10,11], which make the process of extracting root difficult and complex because of the diversity of morphological formulas and the multiplicity of words 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) [12,13]. The second class is formed of statistical methods which are simple, fast, and do not require any morphological rules but some calculations [14,15, 16,17, 18,19,20].

In this paper, we propose a new statistical method which permits to extract roots of Arabic words using the approach of n-grams of characters without using any morphological rule. The paper is organized as follows: the first section is a general introduction to the field of study. The second section presents some related works, so we review some papers that treat the problem of extraction of Arabic word’s roots. In the third section we introduce our new algorithm. The fourth section presents the experiments that we have done to test our new method and also presents the obtained results. In the last section we conclude our work by summarizing our realized work and giving some ideas to improve it in the future.

2. Related Works

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: [9], [21] Khoja’s roots extractor removes the longest suffix and 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 [22,23,24,25]. [13] Propose a linguistic approach for root extraction as a preprocessing 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 texts’ collection consists of eight categories, the author achieved a success ratio about 73.74%. [26] Presents a new Arabic root extractor that tries to assign a unique root for each Arabic word without having an Arabic roots list, a

word patterns list, or the list of Arabic prefixes and suffixes. The algorithm predict the letters 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 of two parts, the first part gives the rules that distinguish between the Arabic definite letter "ـال AL, La" and the original word letters "ـال". 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 root extracting process.

In the second class of algorithms we can note the following: [14] 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 [14] did not explain on what basis did it use such ranking or weighting. [10] Proposes an algorithm to extract tri-literal Arabic roots, this algorithm consists of two steps; in the first step they eliminate stop words as well prefixes and suffixes. In the next step, they remove the repeated word's letters until only three letters are remained, and 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 were very promising and give an accuracy of root's extraction over than 73%. [27] Propose a new way to extract the roots of Arabic words using n-grams technique. They used two similarity measures; 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. [28] propose a new algorithm to find a system that assigns, for every non vowel word a unique root. The proposed system consists of two modules; the first one consists of analyzing 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. They validate their algorithm using NEMLAR Arabic writing corpus that consists of 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. [29] Propose a new algorithm which use the n-grams technique. In this technique, both the word and its assumed root are divided into pairs called bi-grams, then the similarity between the word and the root is calculated using equation (1) [30]. This process is repeated for each root in the roots list:

$$S = 2 \times C/(A + B) \quad (1)$$

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 use also 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 ، نه ، نب ، نو ، نن ، هب ، هو ، هن ، بو ، بن ، ون)
R_1 = "فتح" ➔ (فت ،فح ، تح)
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 unique bigrams in the word *w*
N_{R_i} : The number of unique bigrams in the root *Ri*
N_{WR_i} : The number of common unique bigrams between the word *W* and the root *Ri*
$N_{W\bar{R}_i}$: The number of bigrams belonging to the word w and do not belong to the root *Ri*
($N_{W\bar{R}_i}$=$N_W - N_{WR_i}$)
$N_{R_i\bar{W}}$: The number of bigrams belonging to the root *Ri* and do not belong to the word *w*
($N_{R_i\bar{W}}$= N_{R_i} - N_{WR_i})
For the previous example we have:
N_W=18, 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, $N_{W\bar{R}_1} = 18$, $N_{W\bar{R}_2} = 18$, $N_{W\bar{R}_3} = 15$, $N_{W\bar{R}_4} = 18$, $N_{W\bar{R}_5} = 17$, $N_{W\bar{R}_6} = 17$, $N_{R_1\bar{W}} = 3$, $N_{R_2\bar{W}} = 3$, $N_{R_3\bar{W}} = 0$, $N_{R_4\bar{W}} = 3$, $N_{R_5\bar{W}} = 2$, $N_{R_6\bar{W}} = 2$.

Step3: 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.

Step4: 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 :

$$D(w, R_i) = 2 * N_{wR_i} + k * N_{w\bar{R}_i} + k * N_{R_i\bar{w}} \quad (2)$$

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+15*100+0*100 = 1506
D(w, R_5) = 2*1+17*100+2*100 = 1902
D(w, R_6) = 2*1+17*100+2*100 = 1902

Step5: 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.
3. Valid for strong roots 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 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 corpus which can be classified according their sizes into: small corpus, middle corpus, and large corpus.
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 note also that many of them are vocalic roots which contain at least one vowel.
3. The file of golden roots which contain the correct roots of all words present in our corpus (the file in (1)), this golden list was prepared by an expert linguist and used as reference list, i.e., by comparison between 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).

Table 1. Corpus 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	1500	450	1500

Table 2. An example of morphological forms (gross words).

Word	Word	Word	Word	Word
مأخذ	*أوامر*	*باحث*	*اجتماعات*	*مأخذ*
مؤاخذة	*مؤتمر*	*بحوث*	*اجتماعيات*	*مؤاخذة*
مؤاخذون	*مؤامرة*	*أبحاث*	*جموع*	*مؤاخذون*
مؤاخذات	*متآمرون*	*باحثون*	*جوامع*	*مؤاخذات*
مؤازرة	*يأتمرون*	*باحثات*	*يجمعون*	*مؤازرة*
مأكل	*يأتمرن*	*ابتهال*	*يجمعن*	*مأكل*
أكلات	*أمرهم*	*مبتهل*	*اجتهاد*	*أكلات*

Table 3. An example of trilateral, quadrilateral, quinquelateral, hexalateral roots.

Trilateral roots	Quadrilateral roots	Quinquelateral roots	Hexalateral roots
زرع	أكرم	انطلق	استعمل
صنع	أعان	انكسر	استحسن
تجر	أعطى	احتوى	استعان
جمع	حطَم	اقتصد	اخشوشن
نفر	ربّى	اخضرّ	ادهامّ
طار	حاسب	تحدّى	احرنجم
سعل	طمأن	تنازل	اقشعرّ
صدع	زلزل	تدحرج	اطمأنّ

Table 4. Examples of obtained results when segmenting words into bi-grams.

Word	N-grams Ng.Frequencies	Nb.Ng (N_W)
يتعلمون	يت يع يل يم يو ين تع تل تم تو تن عل عم عو عن لم لو لن مو من ون 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	28
عالم	عا عل عم ال ام لم 1 1 1 1 1 1	6
كاتب	كا كت كب ات اب تب 1 1 1 1 1 1	6
كتاتيب	كت كا كت كي كب تا تت تي تب ات اي اب تي تب يب *2* 1 1 1 *1 1 2 2* 1 1 1 *1*	12
اقتصاد	اق ات اص ا ا اد قت قص قا قد تص تا تد صا صد اد 1 1 1 1 *2* 1 1 1 1 *1* 1 1 1 1	14
يقصدون	يق يص يد يو ين قص قد قو قن صد صو صن دو دن ون 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	15
استخدم	اس ات اخ اد ام ست سخ سد سم تخ تد تم خد خم دم 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	15
سنستدرجهم	سن سس ست سد سر سج سه سم نس نت ند نر نج نه نم ست سد سر سج سه سم تد تر تج ته تم در دج ده دم رج ره رم جه جم هم 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	30
متذبذب	مت مذ مب مذ مب تذ تب تذ تب ذب ذذ ذب بذ بب ذب 1 *2 2 2* 2 *3 1 1* 1	9
متلألئ	مت مل مأ مل مئ تل تأ تل تئ لأ لل لئ أل أئ لئ 1 *2 1* 1 *2* 1 1 *1 1* 2 *1* 1	12
يهزمونهم	يه يز يم يو ين يه يم هز هم هو هن هه هم زم زو زن زه زم مو من مه مم ون وه وم نه نم هم *1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 3 1 1 1 2 1 2*	*23*
المتربّي	ال ام ات ار اب آ اي لم لت لر لب لّ لي مت مر مب مّ مي تر تب تّ تي رب رّ ري بّ بيّ ي 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	28
المربّون	ال ام ار اب آ او ان لم لر لب لّ لو لن مر مب مّ مو من رب رّ رو رن بّ بو بن بّ وّ ن ون 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	28
طائرات	طا طئ طر طا طت ائ ار ا ات ئر ئا ئت را رت ات *1 1 1 1 1 2 1 1 1 1 1 1 2*	*13*

Table 4. Examples of obtained results when segmenting roots into bi-grams.

Root	N-grams Ng.Frequencies	Nb.Ng (N_{R_i})
كلّم	كل لّ كم لّ لم مّ م 1 1 1 1 1 1	6
عالج	عا ال عج ال اج لج 1 1 1 1 1 1	6
قصد	قص قد صد 1 1 1	3
اقتصد	اق ات اص اد قت قص قد تص تد صد 1 1 1 1 1 1 1 1 1 1	10
كتب	كت كب تب 1 1 1	3
علم	عل عم لم 1 1 1	3
عمل	عم عل مل 1 1 1	3
خدم	خد خم دم 1 1 1	3
كمل	كم كل مل 1 1 1	3
كمن	كم كن من 1 1 1	3
خمد	خم خد مد 1 1 1	3
درج	در دج رج 1 1 1	3
ذبذب	ذب ذذ ذب بذ بب ذب 1 1 1 3	4
لألأ	لأ لل لأ أل أأ لأ 1 1 1 3	4
هزم	هز هم زم 1 1 1	3
طار	طا طر ار 1 1 1	3
ربّى	رب رّ ري بّ بيّ يّ 1 1 1 1 1 1	6
عقد	عق عد قد 1 1 1	3
تأتأ	تأ تت تأ أت أأ تأ 1 1 1 3	4

Table 5. Extraction of some Arabic words roots using our new algorithm.

Word	Nearest roots	Nb.Common bi-grams	Distance values	Extracted root	Correct root
يتعلمون	كَلم ، عالج ، علم ، عمل ، كمن	3 ، 1 ، 3 ، 2 ، 1	**2506**، 3202 ، 2806 ، 2902 ، 2704	علم	علم
عالم	كَلَم ، عالج ، علم ، عمل	1 ، 3 ، 3 ، 2	**306**، 606 ، 1002 ، 504	علم	علم
كاتب	اقتصد ، كتب	1 ، 3	**306**، 1402	كتب	كتب
كتاتيب	اقتصد ، كتب ، تأتأ	1 ، 3 ، 1	1402 ، **906**، 2002	كتب	كتب
اقتصاد	قصد ، اقتصد ، عقد	3 ، 10 ، 1	1502 ، **420**، 1106	اقتصد	اقتصد
يقصدون	قصد ، اقتصد ، عقد	3 ، 3 ، 1	1602 ، 1906 ، **1206**	قصد	قصد
استخدم	اقتصد ، خمد ، خدم	3 ، 2 ، 3	**1206**، 1404 ، 1906	خدم	خدم
سنستدرجهم	اقتصد ، خدم ، درج ، هزم	1 ، 1 ، 3 ، 1	**2706**، 3102 ، 3802 ، 3102	درج	درج
متذبذب	كتب ، ذبذب	1 ، 4	**508**، 1002	ذبذب	ذبذب
متلألئ	عمل ، كمل ، تأتأ ، لألأ	1 ، 1 ، 1 ، 3	1402 ، 1302 ، 1302 ، 1006	لألأ	لألأ
يهزمونهم	كمن ، هزم	1 ، 3	*2006* ، ***2402***	هزم	هزم
المتربّي	كلم ، عالج ، اقتصد ، كتب ، علم ، ربّى ، طار	1 ، 1 ، 1 ، 1 ، 1 ، 6 ، 1	3602 ، 3202 ، 2902 ، **2212**، 2902 ، 2902	ربّى	ربّى
المربّون	كلم ، عالج ، علم ، كمن ، ربّى ، طار	1 ، 1 ، 1 ، 1 ، 3 ، 1	2902 ، 3202 ، 2902 ، **2806**، 2902 ، 2902	ربّى	ربّى
طائرات	اقتصد ، طار	*3 ، 1*	***1006*** ، *2102*	طار	طار

Table 6. Obtained results when extracting the words roots.

Corpus	Nb.Roots	Nb.Words	Cor. Results	Wr.Results	Suc.Rate	Err.Rate
Small	25	50	49	1	98,00	2,00
Middle	135	270	253	17	94,07	5,93
Large	450	1500	1358	142	90,53	9,47

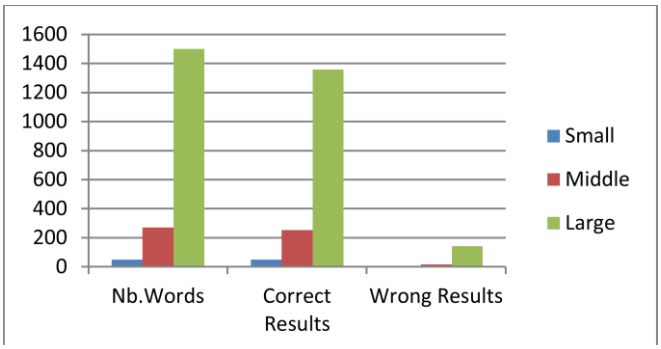

Fig. 1. Correct and wrong results in number of words.

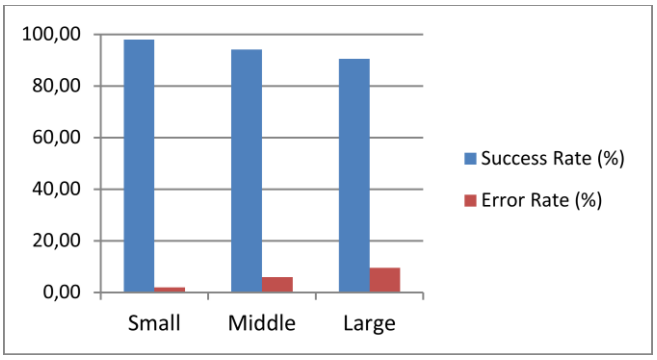

Fig. 2. Calculation of success rate and error rate.

5. Comparison with other algorithms

To show the effectiveness of our proposed algorithm, we concluded our work by establishing a comparison against other known algorithms. For this purpose, we took a sample words list and tried to extract the root of each word using three very known algorithms which are: khodja stemmer, Nidal et al stemmer, and our proposed stemmer, the obtained results are shown in table 7.

In the other hand, we illustrated the obtained results when applying the three above algorithms on the three corpus used in the experimentation, namely: the small corpus, the middle corpus, and the large corpus, and then we summarized the obtained accuracy for each algorithm in table 8.

Table 7. Extraction of some words roots using the three algorithms.

Word	Extracted root			
	Khodja algorithm	Nidal et al algorithm	Our proposed algorithm	Correct 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
المربّي	ربأ	ربّى	ربّى	ربّى
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
نوازل	نزل	تنازل	نزل	نزل

Table 8. Illustration of obtained accuracy for the three algorithms

Corpus	Size		The obtained accuracy (suc_ rate, err_ rate)%					
	Nb.roots	Nb.words	Khodja algorithm		Nidal et al algorithm		Our proposed algorithm	
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	450	1500	73,26	26,74	57,79	42,21	90,53	9,47

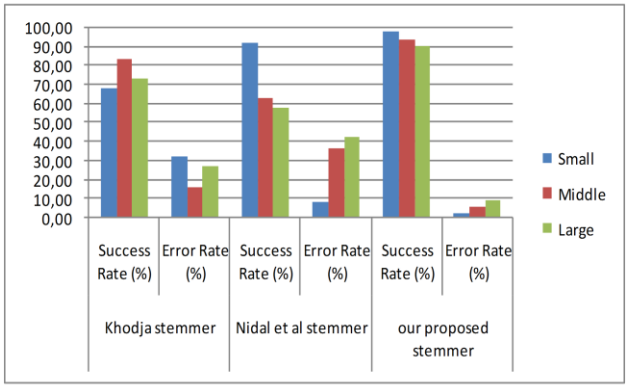

Fig. 3. Comparison between three algorithms

6. Discussion

From table 7, we see that khodja stemmer 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., حواسيب , متألئ , سنستدرجهم) completely a new word and sometimes a wrong word that does not exist in Arabic language (i.e.,*(طور ، طائرات)* , *(ربن ، المربّون)* , *(قوع ، وقيعة)*). The same thing can be said for Nidal et al algorithm although it's gives best results than khodja algorithm, but it fails for many words like : **(***سجد، ناسج)*, *(زنن ، يزنونهم)*, **(**سي، حواسيب**)**. For the same cases, our algorithm gives always the correct root and the failure in our algorithm is very limited.
From Table 8 and figure 3, we can deduce that our proposed algorithm gives the best results for the three used corpus with a very high accuracy. We note here the value 98 % for the small corpus, 94,07 % for the middle corpus, and 90,53 % for the large corpus

7. Conclusion and Perspectives

In this paper we have studied how we can reduce the size of terms in Arabic text categorization by replacing many words by their common root. In this purpose, we exposed the most known algorithms and techniques in the field, Including morphological algorithms mainly based on the use of morphological rules and grammatical patterns 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 and easy to implement on machine, does not require the removal of affixes nor the use of any morphological rules and grammatical patterns, capable to find all types of roots, i.e., trilateral, quadrilateral, quinquelateral, and hexalateral 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 algorithms which are very known in the field, namely: Khodja algorithm, Nidal et al algorithm. The first one fails sometimes in getting the correct root of the given word and for many words it produced one of two results: (1) not stemmed word (2) completely a new word and sometimes a wrong word that does not exist in Arabic. The same thing can be said for second one, although it gives best results than the first, but it fails for many words. For the same cases, our new algorithm gives always 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 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. Fatma, A.H., Keith, E.: Rule-based Approach for Arabic Root Extraction: New Rules to Directly Extract Roots of Arabic Words. Journal of Computing and Information Technology CIT journal, Zaghreb, 57–68. (2014)
2. Ghazzawi, S.: The Arabic Language in the Class Room. 2nd EDN, Georgetown University, Washington DC. (1992)
3. ETHNOLOGUE, http://www.ethnologue.com/statistics/size. accessed 16 January 2014.
4. Al-Kamar, R.: Computer and arabic language computerizing. Dar Al Kotob Al-Ilmiya, Cairo, Egypt. (2006)
5. Ghwanmeh, S., G. Kanaan, R. Al-Shalabi and S. Rabab'ah: Enhanced algorithm for extracting the root of Arabic words. In: Proceeding of the 6th International Conference on Computer Graphics, Imaging and Visualization, Aug. 11-14, IEEE Xplore Press, Tianjin, Chain, pp: 388-391. (2009)
6. Yousef, N., I. Al-Bidewi and M. Fayoumi.: Evaluation of different query expansion techniques and using different similarity measures in Arabic documents. Eur. J. Sci. Res., 43, 156-166. (2010).
7. Wightwick, J. and M. Gaafar. : Arabic Verbs and Essentials of Grammar, 2E (Verbs and Essentials of Grammar Series). 2nd EDN., McGraw-Hill Companies, Inc., ISBN-10: 0071498052, pp: 160 (2007).
8. Al-omari, A., Abuata, B., Al-kabi, M.: Building and Benchmarking New Heavy/Light Arabic Stemmer. In: The 4th International conference on Information and Communication systems (ICICS'13), (2013).
9 Shereen, K., Garside, R.: Stemming Arabic text. Technical report, Computing Department, Lancaster niversity, 1999., [online] available: http://www.comp.lancs.ac.uk/computing/users/khoja/stemmer.ps, last visited 1999.
10. Momani, M., Faraj, J.: A novel algorithm to extract tri-literal Arabic roots. In: Proceedings of the IEEE/ACS International Conference on Computer Systems and Applications. May 13-16, IEEE Xplore Press, Amman, pp: 309-315. (2007).
11. Al shalabi, R.: Pattern-based stemmer for finding Arabic roots. Information Technology Journal, Vol.4, No.1, 38–43. , (2005).

12. Hajjar, A.E.S.A., M. Hajjar., Zreik, K.: A system for evaluation of Arabic root extraction methods. In: Proceeding of 5th International Conference on Internet and Web Applications and Services (ICIW), May 9-15, IEEE Xplore Press, Barcelona, pp: 506-512. (2010).
13. Al-Nashashibi, M.Y., D. Neagu., Yaghi,A.A.: An improved root extraction technique for Arabic words. In: Proceeding of 2nd International Conference on Computer Technology and Development (ICCTD), Nov. 2-4, IEEE Xplore Press, Cairo, pp: 264-269. (2010).
14. Al-shalabi, R., Kanaan, G., Al-Serhan, H.: New Approach for Extracting Arabic Roots. In: Proceedings of the International ArabConference on Information Technology (ACIT'20003), Alexandria, Egypt, pp. 42–59. (2003).
15. Rehab, D.: Arabic Text Categorization. The International Arab Journal of Information Technology, vol. 4, No. 2, 125–131. (2007).
16. Al-Nashashibi, M. Y., Neagu, D., Ali. A. Y.: Stemming Techniques for Arabic Words: A Comparative Study. In: 2nd International Conference on Computer Technology and development (lCCTD 2010), 270–276. (2010).
17. 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, 51-66. (2005).
18. Ghwanmeh S., Al-Shalabi R., Kanaan G., Khanfar K. and Rabab'ah S.: An Algorithm for extracting the Root of Arabic Words. In: Proceedings of the 5th International Business Information Management Conference (IBIMA). Cairo, Egypt. (2005).
19. Mohamad, A., 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, July 2012, Received February 22, 2010; accepted May 20, 2010
20. Al-Shalabi, R., Kanaan, G., Ghwanmeh, S.: Stemmer Algorithm for Arabic Words Based on Excessive Letter Locations. IEEE Conference, (2008).
21. Shereen, K.: Stemming Arabic Text. [online]. Available: http://zeus.cs.pacificu.edu/shereen/research.htm
22. Larkey L., and M. E. Connell. : Arabic information retrieval at UMass in TREC-10.In: Proceedings of TREC 2001, Gaithersburg: NIST (2001)
23. Larkey, S., Ballesteros, L., Margaret, Connell, E.: Improving Stemming for Arabic Information Retrieval: Light Stemming and Occurrence Analysis. In: Proc. of the 25th ACM International Conference on Research and Development in Information Retrieval (SIGIR'02), Tampere, Finland, pp.275–282, (2002).
24. Larkey, S., Ballesteros, L., Margaret, Connell, E.: Light Stemming for Arabic Information Retrieval, Arabic Computational Morphology Text, Speech and Language Technology Vol. 38, 221-243. (2007)
25. Sawalha, M., Atwell, E.: Comparative Evaluation of Arabic Language Morphological Analyzers and Stemmers. In: Proceedings of COLING-ACL. (2008)
26. Hawas, F.A.: Exploit relations between the word letters and their placement in the word for Arabic root extraction. Comput. Sci, Vol 14 , 27-431.
27. Hmeidi, I.I., Al-Shalabi, R., Al-Taani, A.T., Najadat, H., and Al-Hazaimeh, S.A.: A novel approach to the extraction of roots from Arabic words using bigrams. J. Am. Soc. Inform. Sci. Technol., Vol. 61, 583-591. (2010)
28. Boudlal, A., Belahbib, R., Belahbib, A., Mazroui, A.: A markovian approach for Arabic root extraction. Int. Arab J. Inform. Technol, Vol. 8, 91-98.(2011)
29. Yousef, N., Aymen, A.E., Ashraf, O., Hayel, K.: An Improved Arabic Word's Roots Extraction Method Using N-gram Technique, Journal of Computer science JSC. Vol. 10, No. 4 (2014), Published Online (http://www.thescipub.com/jcs.toc).
30. Frakes, W.B.: Stemming Algorithms. In: Information Retrieval: Data Structures and Algorithms, Frakes, W.B. and R. Baeza-Yates (Eds.), Prentice-Hall India, ISBN-10: 8131716929, pp: 131-160. (1992)