Computer Science • 14 (2) 2013 http://dx.doi.org/10.7494/csci.2013.14.2.327

Fatma Abu Hawas

EXPLOIT RELATIONS BETWEEN THE WORD LETTERS AND THEIR PLACEMENT IN THE WORD FOR ARABIC ROOT EXTRACTION

Abstract *This paper presents a new root-extraction approach for Arabic words. The approach tries to assign for Arabic words a unique root without relying on a database of word roots, a list of word patterns or a list of all the prefixes and the suffixes of the Arabic words. Unlike most of Arabic rule-based stemmers, it tries to predict the root-letters positions one by one based on some rules and relations among the word letters and their placement in the word. This paper focuses on two parts of the approach. The first one introduces some rules to distinguish between the Arabic definite article (الāl) and the permanent component (الāl) that may found in any Arabic word. The second one classifies Arabic letters in to groups according to their positions in the word. The proposed approach is a system composed of several modules used to extract the word root. The approach has been evaluated using the Holy Quran words. The evaluation results show a promising root extraction algorithm.*

Keywords rule-based stemmer, word root, suffixes, prefixes, words patterns

1. Introduction

Natural language processing is getting high importance and attention in computer sciences these days. Different applications have been put in to research focusing on this field. Stemming and Root extraction is one of the important applications of natural language processing in which this paper will be focusing on. An efficient Stemmer system is the idea behind the success of many other natural language applications. As a result, researchers start putting many efforts to bring this application in to the light. Many efforts have been deployed recently on the Arabic language. The Arabic stemming process that is based on the language morphological rules is still a difficult task due to the nature of the language itself. Thus, an extensive linguistic analysis of patterns and affixes in Arabic language were carried out prior to this paper. In this work, we propose the first stage of a preliminary version of an Arabic root extractor that depends on two main phases:

The first one handles the words contain component (ال*āl*). In the second phase the position of the letter in the word is taken into account after segmenting the word into three parts.

This paper introduce an answer to the following research question: Can we identify and exploit relations between the word letters and their placement in the word for Arabic root extraction?

This paper is organized as follows: Section 2 includes a brief review of the Arabic language features. Section 3 introduces a background of Arabic language morphology and previous works. Section 4 discusses the proposed approach. Experimental results are found in section 5. Finally, conclusions and discussions of future works are found in Section 6.

2. Overview of Arabic language

The Arabic Language is the official language of 26 states. It counts more than 300 million first language speakers more than that of any other Semitic language [12]. The Arabic Language is a very rich language, in consequence, this makes the morphology process of Arabic words not an easy task. The following are some significant features for the Arabic language:

- Letters are mostly connected. Most letters change form depending on whether they appear at the beginning, the middle, the end of a word, or by their own. For example, the letter (*ġ* غ) has different forms: (*ġ*- غـ) at the beginning of the word as in the word (*ġrb* غرب) which means *"west"*, (-*ġ*- غـ) in the middle of the word as in the word (*yġyyr* يغيّر) which means *"change"*, (-*ġ* غ) at the end of the word as in the word (*ṣmġ* صمغ) which means *"glue"*, and it appears by its own

at the end of the word (فرَاغ*frāġ*) which means *"space"*.

- Arabic letters can be divided into two groups; Affixes letters[1]: those that form the word (سأَلتمونيهَا*s'altmwnyhā*), and the rest which never come as affixes.
- The letters (ا،و،ي*ā,w,y*) are the long vowels and the rest are consonants.
- Short vowels which are letters marked by vowel diacritics, and other special symbols which are sometimes used to avoid the ambiguity of the meaning of that word, so it is easy to check its grammar and to extract its proper root. For example: the word (يعبر*y'br*) can be read as (يَعبرُ*ya'bru*) which means *"to pass"* or as (يُعبّرُ*yu'bbru*) which means *"to express"*.
- The gemination mark (*called AL-Shaddah in Arabic*) is used to indicate a doubled consonant while pronouncing it. For example: the word (كسّر*kssr*) which means the past verb *"smashed to pieces"* consists of three letters, the middle one is doubled. Unfortunately, people do not explicitly mention these marks in writing; instead, they depend on their knowledge in understanding the existence of these marks in the words.
- Words are derived from basic building blocks with tri-consonantal roots. For example the word (استهتَارهم*āsthtārhm*) which means *"their irreverence"* has the following structure: the prefix (است*āst*), the infix (ا*ā*), the root (هتر*htr*), and the suffix (هم*hm*).

For more details about Arabic language see [2].

3. Background and previous works

Morphology is the field of linguistics that studies the internal structure and formation processes of words [3]. A morpheme is often defined as the smallest meaningful and significant unit of language, which cannot be broken down into smaller parts [3]. In Arabic language for example, the word (تركهَا*trkhā*) which means *"he left her"*consists of two morphemes, the verb (ترك*trk*) which means *"left"* and the pronoun (هَا*hā*) which means *"her"*. There are two types of morphemes: the main morpheme which is called the word's stem or root, and the affixes. The root is the base form of the word that gives the main meaning of the word, while the affixes add additional meanings to the word. Stemming is the process of reducing the derived words to their base form.

[1]Letters act as prefixes, infixes, or suffixes in Arabic words.

Many techniques have been developed to process languages such as English [5, 7, 8], and French [9, 10]. However, in 2004, Al-Sughaiyer and AL-Kharashi [1] came up with a classification for stemming strategies for the Arabic language. They divided the stemming strategies in to four approaches, namely, table lookup, combinatorial, rule-based approach (also referred to as linguistic strategy), and pattern-based approach. The rule-based strategy is a commonly applied stemming technique. It is based on the linguistic rules obtained through a detailed analysis of the Arabic morphology system. The rules describe the morphological structure of the words. The linguistic strategy simulates the same process of an expert linguist during his analysis of a given Arabic word. The most common stemmers in this field are: Khoja root-stemmer [4] and Al-Shalabi pattern-base stemmer [11], Momani and Fraj [6].

4. Approach description

This section presents a new approach for extracting the Arabic word roots. The approach is considered a blind approach as it attempts to find the word root without even having a database of word roots, a list of the words patterns or even a list of all the prefixes and the suffixes of the Arabic words. Instead, it tries to predict the positions of the letters that may form the word root by classifying the Arabic letters according to their positions in the word. In this approach each letter is assigned a value either 0 or 1 to mark it as a root letter or not. The values that are assigned to the word letters depend on two things:

- Whether the letter appears as an affix letter or not.
- The placement of the affix letter (at the beginning, the middle or the last part of the word).

The system consists of several phases explained in the following sub-sections.

4.1. Word and letters divisions

In this approach, each word is divided in to three parts (S_1, S_2 and S_3 respectively). The length of S_1 and S_3 are computed by dividing the word length by 3 and then rounding the result to the nearest integer; the length of S_2 is equal to

$$(\text{wordsLength} - (S_1 + S_3)).$$

To illustrate, let $W = (L_1, L_2, L_3, \cdots, L_n)$ be the input word, where L_i is the words letter, and n is the length of the word. Then

1. $W_{(segments)} = S_1 + S_2 + S_3$;where each S_i is a words part.
2. $Length(S_1) = Length(S_3) = Round(n/3)$.
3. $Length(S_2) = n - 2 \times Length(S_1)$.

The approach also classifies Arabic word letters into eight sets or groups explained in section 4.3 table1. During executing the proposed algorithm, each letter in each group could have a weight 0, 1, or -1 to represent its existence in the word root,

(0=OFF, 1=ON, −1=Undetermined). The word root is represented as a list of length n as follows:

$$R = (r_1, r_2, r_3, \cdots, r_n)$$

Where r_i is the letters weight.

For example: the word (سنكتب*snktb*) which means *"we will write"* and its root will be represented as:

$N = 5$;
$W =$ (ب ، ت ، ك ، ن ، س *s , n , k , t , b*) ;
$W_{(\text{segments})} = 2 + 1 + 2$;
$R = (0, 0, 1, 1, 1)$;

At the beginning of the execution, all r_i are set to -1, which means no decision yet has been made about this letter.

4.2. Handling words contain (ال*āl*) component and words with gemination mark

The definite *"Al"* (ال*āl*): is a particle in the Arabic language that is attached with nouns to indicate the type of reference being made by the noun. For example, the word (كتَاب*ktāb*) which means *"book"'* can be made definite by prefixing it with *"Al"*, resulting in (الكتَاب*ālktāb*) which means *"the book"*. Consequently, *"Al"* is typically translated as *"the"* in English.

Unlike most other particles in Arabic, the definite *"Al"* is always prefixed to another word and it never stands alone. The definite *"Al"* is not considered a permanent component of the word to which it is prefixed. It is added and removed to toggle between the definiteness and indefiniteness of the word. The particle *"Al"* may also appear as a permanent component of the word. To distinguish between the definite *"Al"* and the permanent *"Al"*, a definition for a permanent *"Al"* is introduced in this section.

Definition:

We do not consider (ال*āl*) as a definite article, if at least one of the following cases is true.

Let k be the position in the word (ال*āl*) starts at, and suppose i, j *and* m are three other positions where $m > k > j \geq i$ and $m = k + 2$. then several cases are taken into account to consider (ال*āl*) a permanent component of the word it is prefixed.

- Case one:
  - $L_m \in \{the\ sun\_letters\}$,AND $L_{m+1} \notin \{AL\text{-}Shaddah\}$
- Case two: $k = 2$ (one letter precedes ال$\bar{a}l$)
  - $L_j \notin \{$ب، ف، و، ك$k, w, f, b\ \}$ OR
  - $L_j \in \{$ب، ف، و، ك$k, w, f, b\ \}$ AND N<6.
- Case three: $k = 3$ (two letters precede ال$\bar{a}l$)
  - L_i *and* L_j Belong to {ب، ف، و، كk, w, f, b }.
    - $L_i = L_j$
    - $L_i \in \{$ب، ك$k, b\ \}$ AND $L_j \in \{$ب، ف، و، ك$k, w, f, b\ \}$
    - $L_i \in \{$و$w\ \}$ AND $L_j \in \{$ف$f\ \}$
  - ($L_i$or L_j Does not belong to {ب، ف، و، كk, w, f, b })
- Case four: $k > 3$ (more than two letters precede ال$\bar{a}l$).

In case one; (ال$\bar{a}l$) is considered a permanent component of the word when it is directly followed by un-doubled sun letter. In this case L_m is a root letter except for $L_m \in \{$ت$t\ \}$. Case one should be checked up before all the cases following.

Examples of this case: (جَالسين$\check{g}\bar{a}lsyn$), (الزَام$\bar{a}lz\bar{a}m$).

In case two, two things should be tested, the letter preceding (ال$\bar{a}l$), and the length of the word. If the letter preceding (ال$\bar{a}l$) is not one of the prepositions {ب، ف، و، ك$k, w, f, b\ \}$, then(ال$\bar{a}l$) is a permanent component of the word. Otherwise, the length of the word should not exceed five letters. The reason the size of the word is considered in the second condition is to make sure that L_j is not acting as a preposition, since the minimum number of letters the nouns in Arabic may include is three[2]. In this case L_j is considered a root letter.

Examples on this case: (فَالق$f\bar{a}lq$),(بَالغ$b\bar{a}l\dot{g}$),(وَالي$w\bar{a}ly$).

In case three, both L_i and L_j should be one of the prepositions {ب، ف، و، ك$k, w, f, b\ \}$, in which either both are similar or L_i is either {ب$b\ \}$ or {ك$k\ \}$ or L_i is {و$w\ \}$ followed by {ف$f\ \}$. In this case $L_j$is a root letter only when $L_i = L_j$

[2]three letters for the noun+ two letters for(ال$\bar{a}l$)+one letter for L_j

OR $L_j \notin \{$و$w\}$.

Examples on this case: (كوَاليس$kwālys$), (بَالغ$bālġ$).
In the second condition of case three, only one of the two letters is not a preposition {ب، ف، و، كk ,w ,f ,b }is enough to consider(ال$āl$) a permanent component of the word. In this case L_j is considered a root letter except for $L_j \in \{$و$w\}$.

Examples: (موَالي$mwāly$), (خيَالَات$ẖyālāt$), (موَالح$mwālḥ$).

In case four, the component(ال$āl$) is always permanent in the word, the only prefixes found in the holy Quran that negate this case is (أَفبafb). Example: (أَفبَالبَاطل$afbālbāṭl$). In all four cases mentioned above, L_{k+1} is considered a root letter. If (ال$āl$) is not any of the cases mentioned above then the approach considers it a definite (ال$āl$).

Before extracting the word root, each word is checked against(ال$āl$) or (لل-)[3] according to the cases above. If the letter directly followed the definite(ال$āl$) or (لل-) is one of the letters forms the word (سيلهون لـ$sylhwn$), or one of the letters {ب، ف، كk ,f ,b } then this letter is considered a root-letter.

$$(\text{“Definite } AL\text{”}) \text{AND } L_m \in \{\text{س، ي، ل، ه، و، ن، ك، ف، ب}\} \longrightarrow r_m = 1.$$

Arabic has some special marks. One of these marks is called a gemination mark. The gemination is a mark written above the letter to indicate a doubled consonant while pronouncing it. For each word includes the gemination mark:

If the letters preceded the one with the gemination mark are (ال$āl$) or (لل-)then remove the gemination mark. For example, the word(السّلَام ال$ālsslām$), which means *“the peace”* becomes (سلَام$slām$). Otherwise, remove the gemination mark and consider the letter a root letter. Except for the letter (يy) when it is located in S_3.

4.3. Arabic letters classification

This approach classifies the Arabic letters into eight sets or groups as shown in table1. G1 includes all the letters that form the original letters for any word. The rest of the

[3]When (ال ل) preceeded by the prepositionل, we end up writing ال as لل.

groups include the common augmented letters that may found in Arabic words which consist of the letters that form the word (سألتمونيهَا*s'altmwnyhā*) in addition to the letters (ب، ف، ك*k, f, b*) that may be added at the beginning of the Arabic words as prepositions, for example the letter(ف*f*) could be added to the word (تعمل*t'ml*) to form the word (فتعمل*ft'ml*), the letter (ب*b*) can be added to the word (سرعة*sr'h*) to form (بسرعة*bsr'h*) and the letter(ك*k*) can be added to the word (وسيلة*wsylh*) to form (كوسيلة*kwsylh*). Some of the augmented letters may act as basic letters of the word depending on their position in the word. According to that, we classified those letters in to seven groups (G2 to G7). G2 and G3 contain letters that never act as prefixes when they appear in the first part of the word, G4 and G5 include the letters that never be infixes in the middle part of the word, letters that never act as suffixes are found in G6 and G7, and finally, the last group G8 consists of letters that are always found as Affixes.

Table 1
Arabic word letters classification according the proposed algorithm.

Groups Name	The letters	Location	Weight
G1	{ ث، ج، ح، خ، د، ذ، ر، ز، ش، ص، ض، ط، ظ، ع ، غ، ق}	Anywhere in the word	1
G2	{ه}	First letter of S_1	1
G3	G2 + {ف }	S_1	1
G4	G3 +{ ب، ك، ن، ل ، أ ، ىء ، وء ، ء }	First letter of S_2	1
G5	G4 + {س، م}	S_2	1
G6	{ب، ف، س، ل، و، أ ، ىء ، وء}	Last letter of S_3	1
G7	{ب، ف، س، ل، م، أ ، ىء ، وء ، ء}	S_3	1
G8	{ا}	S_1	0
	{ة}	the end of S_3	0

Algorithm 1 is used to detect the root's letters according to the proposed classification.

Algorithm 1 Arabic Letters Classification

$CountON \leftarrow CountOFF \leftarrow 0$
$Flag \leftarrow true$
$r_1\_Flag \leftarrow false$
$r_n\_Flag \leftarrow false$
$W \leftarrow input\ word$
$N \leftarrow word\ length$
$R \leftarrow \{-1\}$
Apply "identifyAL" routine
$n \leftarrow N$
$W_(segments) \leftarrow S_1, S_2, S_3$
Manipulate the diacritic *AL-Shaddah* in W
FOR (each letter $L_i \in G1$)
    $r_i \leftarrow 1$
    Increment (CountON)
EndFOR
FOR (each letter $L_i \in G8$)
    search S_1, S_3.
    $r_i \leftarrow 0$.
    Increment (CountOFF)
EndFOR
IF ($CountON \geq 3$)
go to 32
EndIF
FOR (each letter $L_i \in G2, \cdots, G7$)
    $r_i \leftarrow 1$
    Increment (CountON).
EndFOR
Detect any changes
IF (Changes)
go to 21
EndIF
Stop and output the root.

5. Evaluations and discussion

Holy Quran words are used for evaluation, a preprocessing module which does the following is applied on a file consists of all the 114 Chapters of the Holy Quran:

- Remove from the texts all the numerals and symbols found in the Holy Quran, punctuation marks, assimilation marks, short vowels, function words, and diacritics except the gemination mark.
- Split the text in to tokens.
- Exclude the stop words.
- Remove duplicate words.
- Save the remaining words in a file.

The file produced consists of 14067 unique words. The length of these words is ranging from 2 to 13 letters. To evaluate the experimental system, this is done using two phases:

- Evaluating the rules of the (ال) component.
- Evaluating the classification of the Arabic letters.

All the words in the file are being used and the experimental results are being tested by matching them against the roots file[4]. The roots file contains the positions of the roots letters of each word in the words file in addition to the roots letters that are missing from the words (if any). Table 2 illustrates the contents of the roots file.

Table 2
Description of the roots file format.

Word	Contents of the Root's file	Description
وَاقعدوَا	3,4,5	The third, fourth and the fifth letters are forming the root(قعد)
وَاعتصموَا	3,5,6	The third, fifth and the sixth letters are forming the root(عصم)
وقنَا	1,2, {ي}	The first two letters and the letter (ي) are forming the root(وقي)
وصدّ	2,3,4	The diacritic (*AL-Shaddah*) is part of the root (صدد)

5.1. PHASE 1: Evaluating the rules of the AL component

In this phase, 1937 words out of 14067 words contain the component AL either as the identity (ال*āl*) as in (وَالنبيّين*wālnbyyyn*) or as a permanent component as in (وَالدي*wāldy*), the ال*āl* procedure rules defined in section 4.2 were successfully able to identify1932 (99.74%) words correctly, and only 5 words were wrong. The procedure only failed in classifying ال*āl* correctly in some words of length 6 that contains one letter preceding the AL component such as: (بَالغيه*bālġyh*) and (وَالدتك*wāldtk*) This phase was also able to find some roots letters of some words. Mention that the entire

[4]We selected the roots letters of each word by using the reliable Arabic dictionary Muktar Al_Sahah

root letters were detected in this phase was correct. Table 3 and table 4 show the experimental results after applying the AL procedure and some of the tested words in this phase respectively.

Table 3
The experimental results of phase 1.

Words contain AL	Total Correct Results	Number of words root letters were detected in	
1937	1932	711 (36.8%)	
		Two root letters : 72	One root letter: 639
Percentage	(99.7%)	(3.7%)	(33.1%)

Table 4
Some of tested words of phase 1.

Word	After Phase 1	Roots letters after Phase	Word	After Phase 1	Roots letters after Phase
بَالغه	بَالغه	بل	فَالتقَى	فَالتقَى	ل
بَالغوه	غوه		فَاليوم	اليوم	ي
بَالغيه	غيه		كَالحون	حون	
بَالفتح	فتح	ف	فَالوَالدين	وَالدين	ول
صَالح	صَالح	صل	كَالعهن	عهن	
صَالحين	صَالحين	صل	لبَالمرصَاد	مرصَاد	
ضلَال	ضلَال	ل	العَالمين	عَالمين	ل
لوَالديه	لوَالديه	ل	وَالوَالدَات	وَالدَات	ل
وَالصَالحين	صَالحين	ول	وبَالوَالدين	وَالدين	ول

5.2. PHASE 2: Evaluating the classification of the Arabic letters

This phase experiments the classification of Arabic word letters mentioned in section 4.3, the experimental system analyzed 13856 and failed on 211. The generated roots letters of each word are compared to the ones stored into the roots file taking in to account that the system is in it's first stage.

If any match is found(either a whole match or sub match), then the root analysis

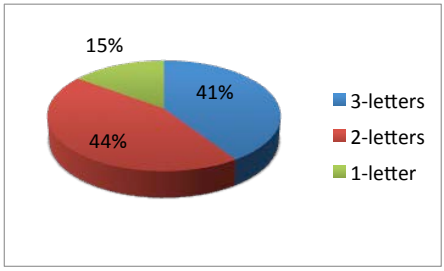
is considered correct, on the other hand, if at least one letter produced by analyzing the tested word is wrong, the root analysis is considered incorrect. By using this method, 13193 results were considered correct, that is about 93.79% and only 663 of the results that are 4.71% of the experimented words are incorrect. The percentages along with the values of the correct results are shown in detail in Table 5 and Figure 1. Some of the tested words after applying phase 2 are found in Table 6.

Table 5

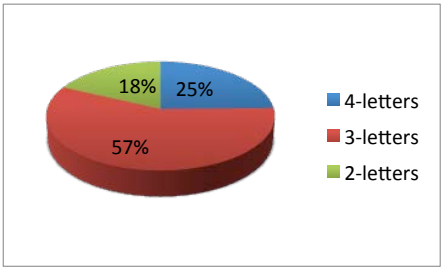
The experimental results of phase 2.

	Total words	**Total correct results**	**Number of correct letters being found**		
Trilateral roots	13974	13103 (93.7%)	3-letters: 5337 (40.7%)	2-letters & one is missing: 5819 (44.4%)	1-letter & two are missing: 1947 (14.9%)
Quadrilateral roots	92	89 (96.7%)	4-letters: 22 (24.7%)	3-letters & one is missing: 51 (57.3%)	2-letters & two are missing: 16 (18%)
Pentliteral roots	1	1	5-letters: 1		
Total	14067 (100%)	13193 (93.79%)			

For the case of the proposed analyzer, it was written in C++ Language. Further, the analyzer was able to derive the roots letters of 14067 words per second on a Compact 2.0 GHz machine with 256 MB of RAM running Windows XP.



(a) Trilateral word roots.



(b) Quadrilateral word roots.

Figure 1. The experimental results of phase 2.

Table 6
Some of tested words of phase 2.

Word	Word root	Root letters detected	Actual root letters	Word	Word root	Root letters detected	Actual root letters
بمزحزحه	زحزح	4	4	العنكبوت	عكب	3	4
حصحص	حصحص	4	4	القنطرة	قطر	3	4
لجبريل	جبرل	4	4	اختصموَا	خصم	3	3
وَالضفَادع	ضفدع	4	4	استأَجره	أَجر	3	3
وزلزلوَا	زلزل	4	4	استأَذنوك	أَذن	3	3
وغرَابيب	غربب	4	4	استكبرتم	كبر	3	3
ابتدعوهَا	بدع	3	3	الأَحَاديث	حدث	3	3
اتّبَاع	تبع	3	3	المستأَخرين	أَخر	3	3
اثَّاقلتم	ثقل	3	3	المقبوحين	قبح	3	3
استأَجرت	أَجر	3	3	المقتسمين	قسم	3	3
استضعفوني	ضعف	3	3	ابتغَاء	بغ	2	3
استقرّ	قرر	3	3	ابيضّت	بض	2	3
اكتسبن	كسب	3	3	المقسطين	قط	2	3
الأَطفَال	طفل	3	3	الملعونة	لع	2	3
الضعفَاء	ضعف	3	3	الرَاسخون	رخ	2	3
السّيَّارة	سير	3	3	المنَافقون	فق	2	3
اتّبعني	تبع	3	3	السّميع	سع	2	3
اقتتلوَا	قتل	3	3	الجزية	جز	2	3
الخرطوم	خرط	3	4	الأَصوَات	ص	1	3

6. Conclusion and future work

A research question was asked at the beginning of this paper: Can we identify and exploit relations between word letters and their placements in word that can be used for Arabic root extraction?

To answer this question an extensive linguistic analysis of patterns and affixes in the Arabic language were carried out. As a result of the analysis, a first stage of a new approach is introduced and showed that we can find relations between the word letter and its placement in the word. The results of the evaluation of this stage showed a promising root extraction algorithm. Although the proposed approach is still a new one, it shows significant results in handling the word root. We would obtain better results if more relations were put under experiment. Our future plan is to extend the algorithm to cover the relations among the Arabic letters, to handle Arabic irregular form words such as weak and two-letter words, and use a larger corpus to test the accuracy of the approach.

Acknowledgements

The author is thankful to Dr. Keith Emmert, Associate professor of mathematics, Tarleton State University, for his valuable comments and help in editing this paper.

References

[1] AI-Sughaiyer I. A., Al-Kharashi I. A.: Arabic morphological analysis techniques: A comprehensive survey. *Journal of the American Society for Information Science and Technology*, 55(3):189–213, 2004.

[2] Duwairi R.: Machine learning for Arabic Text Categorization. *Journal of the American Society for Information Science and Technology (JASIST)*, 57(8):1005–1010, 2005.

[3] Jurafsky D., Martin. J. H.: *Speech and Language Processing: An Introduction to Speech Recognition.* Natural Language Processing, and Computational Linguistics, and Speech Recognition, Prentice-Hall, 2007.

[4] Khoja S., Garside R.: *Stemming Arabic text.* Technical report, Computing Department, Lancaster University, 1999.

[5] Krovetz R.: Viewing morphology as an inference process. In *Conference on Research and Development in Information Retrieval*, pp. 191–202. In Proc. of the Sixteenth Annual International ACM SIGIR, 1993.

[6] Momani M., Faraj J.: A novel algorithm to extract tri-literal arabic roots. In *International Conference on Computer Systems and Applications (AICCSA)*, pp. 309–315. In IEEE/ACS, May 2007.

[7] Paice. C. D.: Another stemmer. *SIGIR Forum*, 24(3):56–61, 1990.

[8] Porter M. F.: An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[9] Savoy J.: Stemming of French words based on grammatical categories. *Journal of the American Society for Information Science*, 44(1):1–9, 1993.

[10] Savoy J.: A stemming procedure and stop word list for general French corpora. *Journal of the American Society for Information Science*, 50(10):944–952, 1999.

[11] Shalabi R. A.: Pattern-based stemmer for finding Arabic roots. *Information Technology Journal*, 4(1):38–43, 2005.

[12] Wikipedia: Arabic language.
`http://en.wikipedia.org/wiki/Arabic_language`, 2013. Online; accessed 18-January-2013.

Affiliations

Fatma Abu Hawas
Faculty of Information Technology and Computer Sciences, Yarmouk University, Irbid, Jordan. e-mail: `fatmih@yu.edu.jo`

Received: 19.12.2012
Revised: 24.01.2013
Accepted: 11.02.2013