Arabic Text Root Extraction via Morphological Analysis and Linguistic Constraints

Amal Alsaad

Department of Electronic & Computer Engineering
Brunel University
London, UK
amal.al-saad@brunel.ac.uk

Maysam Abbod

Department of Electronic & Computer Engineering
Brunel University
London, UK
maysam.abbod@brunel.ac.uk

Abstract—Arabic language is vastly inflected, thus the process of effective Arabic text analysis with correct stem and root extraction is challenging. In this paper we present a linguistic root extraction approach that is composed of two main phases. In the first phase we handle removal of affixes including prefixes, suffixes and infixes. Prefixes and suffixes are removed depending on the length of the word, while checking its morphological pattern after each deduction to remove infixes. In the second phase, the root extraction algorithm is developed further to handle weak, hamzated, eliminated-long-vowel and two-letter geminated words as there is a rationally great amount of irregular Arabic words in texts. Before roots are extracted, they are checked against a predefined list of 3800 trilateral and 900 quad literal roots. Series of experiments has been conducted to improve and test the performance of the proposed algorithm. The obtained results revealed that the roots are extracted correctly has improved comparing with Khoja's stemming algorithm.

Keywords—Arabic root extraction; morphological analyser; natural language processing; data mining; text mining

I. INTRODUCTION

The rapid increase of digitalized textual data has raised the demand for text mining and natural language processing tools and methodologies, to represent these data in an efficient way as possible. This requires feature extraction and different implementations of text processing algorithms for representing the data as required. Text mining methods and algorithms are used in many different information retrieval systems such as search engines, clustering, classification, and other text mining systems.

Arabic language is the 5th amongst the most used languages around the world, yet because of its complex morphological structure there are no available standard Arabic text mining and morphological analysis tools until recently [1]. However, many studies have been conducted to get efficient stemming results in Arabic Information Retrieval systems [2].

In Arabic, both orthography and morphology lead to a great amount of lexical variations where one given word can occur among a large number of dissimilar forms [3]. This would enlarge the indexing structure volume and reduce the performance of the system. Another difficulty in Arabic morphological analysis arises because of the different letter

forms of the tri-literal verbs. Verbs, which have tri-literal roots, are categorized to sound and unsound verbs [4]. Sound (صحيح) verbs are those which does not include a long vowel or hamza in them such as (كتب). Unsound verbs are then divided into weak and compromising. Weak (معتل) verbs have a long vowel or more as part of its original letters such as (قال). Compromising verbs are also divided to hamzated (مهموز), which include a hamza, such as (قرأ), and geminated (مضعف), where the second original letter is doubled such as (جن). In some cases, weak verbs are written with a long vowel that is different from the one of their root following specific Arabic linguistic rules. In other cases, long vowels in verbs are deleted depending on the tense of the verb, becoming eliminated-long-vowel verbs. These cases represent about 30% of the Arabic text [5]. Yet the majority of Arabic stemmers lack the capability of handling these cases. In Fig. 1, the popularity percentage in Arabic text of different verb cases is presented [5].

In this paper, we introduce an enhanced root algorithm that handles the above cases, as well as affixes removal via implementing morphological analysis techniques and specific linguistic rules. Series of experiments and testing were carried out to test the performance and accuracy of the presented algorithm.

Figure 1. Tri-literal Root Types Popularity in Arabic Text

II. ARABIC LANGUAGE COMPLEXITY

Arabic language complexity including its orthography and morphology made it challenging to find a standard Arabic text mining algorithms and tools. In the following we elaborate on these challenges with associated examples where possible.

A. Orthography

Orthography in Arabic is less ambiguous and more phonetic with the use of diacritics. For example, a word can be written using the same characters and be pronounced differently. The main purpose of diacritics including vowel marks, known as *harakat* (حركات), is to provide a phonetic aid to show the correct pronunciation. Arabic vowel marks include *Fatha* (فتحة), *Kasra* (كسرة), *Damma* (ضمة), *Sukun* (سكون), *Shadda* (شدة) and *Tanwin* (تنوين). The pronunciation of these vowel marks are represented in Table I below. However, in Modern Standard Arabic (MSA), vowel marks are not usually included in printed and electronic text, and the understanding and correct pronunciation of the word is determined within its context by the reader.

B. Word Meanings

One word could have several meanings depending on its position and context, despite it having the same pronunciation. For example, as shown in Table II, the Arabic word (قلب) has three meanings as a noun.

C. Variations of Lexical Category

One word can belong to more than one lexical category depending on its meaning and context. Lexical categories include nouns, verbs, adjectives and more. In Table III, we give an example of the word (عين), belonging to different lexical categories depending on its meaning [1].

TABLE I. ARABIC DIACRITICS

Double Constant	No Vowel	Tanwin			Vowel		
بْ /bb/	بُ /b/	بٍ /bin/	بُ /bun/	بُ /ban/	بِ /bi/	بُ /bu/	بَا /ba/

TABLE II. MEANINGS OF THE WORD (قلب)

Word Meaning	Sentence
core	في قلب الأحداث
heart	أجرى عملية قلب مفتوح
center, middle	الكرة في قلب الملعب

TABLE III. LEXICAL CATEGORIES OF THE WORD (عين)

Word Meaning	Word Category	Sentence
Ain	Proper-Noun	عين جالوت
wellspring	Noun	عين الماء
eye	Noun	عين الإنسان
delimitate/be delimitate	Verb/Passive Verb	عُيِّنَ وزيراً للخارجية

D. Morphological Characteristics

Words in Arabic can be formed of a stem alongside affixes and clitics. The stem is composed of a consonant root (جذر صحيح) and a pattern morpheme. The affixes consists of inflectional markers which determines the gender, tense or/and number, while clitics can be propositions (حروف جر), conjunctions (حروف عطف), determiners (محددات), possessive pronouns (ضمائر ملكية) and pronouns (ضمائر).

Morphemes in Arabic are mostly identified by three consonant letters which for the root of the word, as well as several affixes which could be added to the root to form a word. For instance, given the root *kth* (كتب), which is the root of writing, we could inflect various number of words related to the concept of writing such as (كَتَبَ – wrote), (كتاب – book), (كُتِبَ – books), (يَكْتُبُ – he writes), (كاتب – writer), (مكتبة – library) and more.

In addition, the translation of one Arabic word in English can sometimes be composed of a number of words in English. For example, the Arabic word (وبتأثيرها) means (and by her influence). Therefore, segmentation of Arabic textual data is more difficult than it is in Latin languages.

Also, one root can be used to form several words that have different meanings which are not closely related to each other. For instance, the root (علم) can form several words of different meanings when adding affixes as seen in Table IV. Another example of morphological variation is of the word (ذهب) meaning (go), where different clitics are added in different tenses depending on the gender and number of subjects, is shown in Table V.

E. Dual Root

Some words can be formed from more than one root such as the word (رياض), which is derived from the roots (راض) (راض) and (ريض) [2].

F. The Exchange Process

The exchange process (الإبدال) depends on a phonetic rule instead of a syntactic one. For example, the word (قيام) of the pattern (فعل) is derived from the root (قام). But from the consonants in the pattern the extracted root would be (قيم) where the letter 'ي' should be exchanged to 'ا'. The exchange process mostly occurs with the vowel letters (ا، و، ي), but can also arise with other letters as in the word (الصراف) which is changed to (الصراف).

G. Deleted Letters of Words

In some cases, a letter in the pattern of the word is deleted affecting the process of root extraction, like in the word (رأى) of the pattern (فعل). In the present tense of the word, which follows the pattern (يفعل) the letter 'ا' is deleted, and the word becomes (يرى) not (يرأى) for morphological reasons. So the letter 'ع' of the pattern is deleted, becoming (يفل) instead of (يفعل) [14].

TABLE IV. VARIATIONS AND MEANINGS OF THE WORD (علم) WHEN ADDING AFFIXES

Meaning	Suffix	Infix	Prefix	Word
scientific	ية	-	-	علمية
taught us	تنا	-	-	علمتنا
his science	هـ	-	-	علمه
scientists	اء	-	-	علماء
scientist	-	ا	-	عالم
teaching	-	ي	ت	تعليم
teacher	-	-	م	معلم
sciences	-	و	-	علوم
informative	ية	ا	است	استعلامية
information	ات	و	م	معلومات

TABLE V. ARABIC DIACRITICS

Time	Gender of Subject/s	# of Participants	Verb
Past	Male	1	ذهب
Past	Female	1	ذهبت
Past	Male	2	ذهبا
Past	Female	2	ذهبتا
Past	Male	3 or more	ذهبوا
Past	Female	3 or more	ذهبن
Present	Male	1	يذهب
Present	Female	1	تذهب
Future	Male	1	سيذهب
Future	Female	1	ستذهب
Future	Male	3 or more	سيذهبوا
Future	Female	3 or more	سيذهبن

H. Ambiguous Words

Some words in Arabic starts with the letter 'و' or 'ب'. These words are ambiguous when it comes to Arabic data mining as these letters can be part of the original root like in the words (وجود) and (بحوث), while in other words they maybe a prefix like in the words (بإسألهم) and (وأمرهم).

III. RELATED WORK

Many methods and algorithms were developed for text representation in the fields of natural language processing and information retrieval for Arabic language. The main two approaches consist of light stemmers, and root-based stemmers [6]. Light stemmers are employed mainly in information retrieval, where the main concept is to eliminate prefixes and suffixes from a word, generating a stem. In that way, the ideal forms of representative indexing for words is derived [7]. In Fig. 2 is an example of the general steps of Arabic light stemming algorithms [1]. The reason why light stemmers are not concerned with root extraction is that words variants do not always have the same meaning even if they were generated from the same root [3][8]. For example,

the following Arabic words (المكتبة، الكاتب، الكتاب) means (the library, the writer, the book) respectively, despite them belonging to the same root (كتب). Thus, light stemmers intend to improve reduction of feature/keyword whilst maintaining the meaning of the word.

1. Normalize word
 - Remove diacritics
 - Replace ا، إ، أ with ا
 - Replace ة with هـ
 - Replace ى with ي
2. Stem prefixes
 - Remove prefixes: و، لـ، فـ، كـ، بـ، دـ، تـ
3. Stem suffixes
 - Remove suffixes: ها، ان، ات، ون، ين، ية، هـ، ي

Figure 2. Steps of Arabic Light Stemming Algorithms.

The second approach is root-based stemming, where roots of the words are extracted by defining morphological analysis techniques. As the roots are extracted, the words are then grouped accordingly [6]. The fundamental two steps in root-based stemmers are to firstly remove prefixes and suffixes, and to secondly extract the roots by analyzing the words depending on their morphological components. That is usually achieved by identifying rule-based techniques, patterns table lookup, or by a combination of both. Root based stemmers take in account that Arabic complex morphology leads to a great amount of lexical variation, and as mentioned previously, this would enlarge the indexing structure volume and reduce the performance of the system. Another cause why root-based stemmers are used is that words entered as user query in information retrieval systems does not exactly match those included in the relevant documents [2].

One of the earliest and most well known techniques developed for Arabic root-based stemmers is Khoja's stemming algorithm [9]. Khoja's stemmer eliminates the longest suffix and the longest suffix and prefix, and then matches the rest of the word against a list of verbal and noun patterns to extract the root. Another step is performed to check the correctness of the root by checking it against a list of roots. If the extracted root is found, it is then preserved as the root of the word. The stemmer also utilizes a number of linguistic data files, including lists of all diacritic characters, punctuation characters, definite article, and stop words. The stemmer also handles some cases of Arabic tri-literal words that are weak, hamzated, geminated or eliminated-long-vowel. But the algorithm has a number of weaknesses. Firstly, the word (منظمات) which means (organizations) is stemmed to the root (ظما) which means (he became thirsty) instead of the correct root (نظم). Another issue is that when the word is deducted to a tri-literal word, the weak letter is deleted in the first place, and then the last letter is doubled, or another weak letter or an *alif* is added to

the word. That leads to extracting a root that is of another word, which is not related to the word. For example, the extracted root of the word (روايات) is (روي), where the correct root is (روي). As well, the extracted root for the word (آخر) is (خرر), where the correct root is (أخر).

Al-Shalabi, Kanaan, and Muaidi have developed a root extraction algorithm for tri-literal roots, which does not make use of any dictionary [10]. The algorithm counts on giving weights to a word letters, for each letter, the weight is multiplied by the position of it. Consonant letters were weighted of zero, where different weight values were assigned to the letters in the word (سألتونيها), as affixes are formed by a mixture of these letters. Specific computations are then performed over these weights to extract the correct root.

Another similar stemming algorithm for Arabic trilateral words is Al-serhan stemmer [11]. It employs Back Propagation Neural Network to extract roots from five letters Arabic words. Four types of input were generated encoded with binary digits, one relates to the original letters, while the other three classify the letters group of the word (سألتونيها) depending on their occurrence frequency as an affix letter.

Sawalhi and Atwell evaluated different Arabic morphological stemmers and analyzers [12]. They disclosed that Khoja stemmer has attained the highest accuracy rate over the trilateral root extraction algorithms and other stemmers. However, the majority of Arabic words, between 80 and 85%, are derived from trilateral roots while the rest are derived from quad-literal, penta-literal or hexa-literal roots[13]. Khoja stemmer achieved the highest accuracy rate as it works efficiently for tri-literal and while stemming quad-literal roots as well.

IV. PROPOSED ALGORITHM

In this work, we present a new root extraction algorithm that is composed of three main phases. These phases are processed after a text preprocessing stage where all stop words and vowel marks are removed. In the first phase we focus on eliminating suffixes and prefixes according to the length of the word, while employing a pattern matching process to remove infixes and extract the root of the word. The words are matched against patterns of similar length after every prefix/suffix deletion, to improve the speed of root extraction and avoiding removing original letters of the word that are equal to a group of a suffix/prefix letters. In the second phase, if the word root is still not found, it is decided to remove suffixes and prefixes that are of one letter where the word is more than three letters long. If the word is three letters long, it is then processed depending on it being hamzated, weak, geminated, or a word with eliminated long vowel. Finally, if the word is of two letters, it is processed depending on its being a geminated or a long-vowel-eliminated word. Below is a detailed explanation of the three phases of the algorithm.

A. Phase 1

Within this phase, the algorithm is defined to process words according to their length, starting with rules for long words and moving towards shorter words. After every suffix/prefix deletion the word is checked against a list of patterns of the same length. As seen in Table VI. If a pattern is matched, the root is extracted and is validated by checking if it exists in a predefined root list of 3800 trilateral and 900 quad literal roots.

- First, if the word starts with the letters 'ال' then remove them.
- If the word length is equal or greater than six, check for the following prefixes and remove them:
Prefixes- كال، بال، فال، مال، وال، ولل، است، يست، تست، مست
- If the word length is still greater than or equal to five, remove the following prefixes/suffixes:
Prefixes- سن، سي، ست، لي، لن، لت، لل
Suffixes- ون، ات، ان، ين، تن، تم، كن، كم، هن، هم، يا، ني، تي، وا، ما، نا، ية، ها، اء
- If the word is equal or greater than four letters long, remove the following prefixes/suffixes:
Prefixes- ت، ي، ب، ل
Suffixes- ت، ة، ه، ا، ي
- If no root was found, the word is then processed through the second phase of the algorithm.

TABLE VI. ARABIC PATTERNS AND ROOTS

Length of Patterns/Roots	Patterns
Length 4	فاعل، فعال، فعيل، فحول، أفعال، مفعال، فعلة، فحل
Length 5 patterns of tri-literal roots	تفعيل، تفعلة، مفاعل، أفعال، متفعال، تفاعل، تفوعل، انفعال، افتعل، مفتعل، فعالن، فعالل فعالء، مفعال، فواعل، فعية، فعائل، أفعال، مفعول، تفعيل، فاعول، فعلي، فعالة، تفعلة، مفعلة، أفعلة
Length 5 patterns of quad roots	تفعال، مفعال، فعلة، فعائل
Length 6 patterns of tri-literal roots	استفعال، مستفعال، افتعال، انفعال، متفاعل، مفاعلة، مفاعل، أفعلاء، أفاعيل، افعوعل، مفعوعل
Length 6 patterns of quad roots	فعائل، افعال، افتعل، متفعال
Length 6 or more	استفعال، افعيعلل

B. Phase 2

In this phase the cases of hamzated, weak, geminated and eliminated-long-vowel words are handled.

- If the word contains one of the hamzated letters 'ؤ', 'و', 'ي' or 'ء', such as in the word (يؤكل), or the letter 'أ' which expresses the hamzated *alif* 'أ' with the long vowel *alif* 'أ' as in the word (مأرب), change it to 'أ' then validate if it was a root or not. In the case of (يؤكل), the letter 'ي' is removed in Phase 1 and the letter 'ؤ' is changed to 'أ', giving us the correct root (أكل).

- If the word's second letter is weak, 'ا', 'ي' or 'و', then change it to 'و', if the root was not valid change it to 'ي', if the root was not found change it to 'ا'. An example of this case is the word (قال) which root is (قول).
- If the root is still not found, that mean the word is either geminated or an eliminated-long-vowel word with one letter prefix/affix. In this algorithm we remove the prefixes of the letters 'ي' and 'ت' like in the words (يجن) which is geminated, and the word (نقم) which is an eliminated-long-vowel word.
- If the root was not found and the word is three letters long, return root was not found, otherwise if the word is two letters long, proceed to Phase 3.

C. Phase 3

In this phase, words that are two letters long are handled. These words can either be geminated, eliminated-long-vowel or hamzated with an *alif* that is removed from imperative verbs like in (كل) and (خذ) of the roots (أكل) and (أخذ).

- The first step here is to double the last letter of the word that is geminated, as doubled verbs roots have the second highest percentage of popularity in Arabic language after consonant verbs [5]. An example of this case is the verbs (صد) and (مد) of the roots (صدد) and (مدد).
- If the root is still not valid, the word could be an eliminated-long-vowel. Thus we add the long vowel 'و' in between, as weak roots of the vowel 'و' comes third in the roots popularity list.
- If the root was not found, add the vowel 'ي' in the middle of the word.
- Lastly, if the root is still not found, add a hamzated *alif* in the beginning then check for root validation. If the root was not found at this stage, return root was not found.

V. EXPERIMENTS AND RESULTS

A. Data Set

In order to support and test our algorithm, we selected a number of entries from Al-Sulaiti's online Arabic corpora [15]. The data set of Al-Sulaiti is collected to provide a prototype text material for Arabic teachers, new learners and mainly Arabic language researchers and engineers. The corpus consists of 842684 words and 14 different categories. In previous work on Arabic root extraction, most testing methods does not include manual checking to verify if the root of the word was extracted correctly and does actually belong to that word or not. Instead, the roots are defined as correct if the word was shortened to a tri-literal word, or if it did exist in a predefined list of roots. Also, the percentage of the correctly extracted roots is shown to be higher than other compared algorithms within the work despite using a different data set of different amount. Thus, we decided to

manually verify the results of the algorithm, selecting several entries making up to 4341 words as the total text, to be compared with Khoja's stemming algorithm result.

B. Testing and Evaluation Method

Arabic stemming and root extraction research included various different algorithms, but only a few has focused on solving the problem of tri-literal words that are weak, hamzated, geminated or eliminated-long-vowel. Nevertheless, Khoja's stemmer is one of the very well known Arabic stemming algorithms that also handle these cases. Therefore, we process our data through our root extraction system taking the text input from a text file containing the data set. The same data set is then processed through Khoja's stemming system which is available for download [9]. After that, the results are manually checked and compared for evaluation.

C. Results and Findings

Using the same collected data set as input to both our root extraction system and Khoja's system, we achieve the results as seen in Table VII and Fig. 3 below. It can be seen that Khoja's system extracted 3162 roots out of 4341 words (73%), while our system has extracted 3061 roots (70%). This is because of excessive root extraction steps in Khoja's algorithm that leads to extracting roots for Arabicized and Proper noun words, and words that are a combination of a prefix/suffix and a stop word. For example, the system extracts the root (طلي) for the word (إيطاليا), which means the country Italy, and the root (ولي) for the word (ولا), which is a combination of the prefix *waw* (و) and the stop word (لا). The number of inaccurate roots extracted is also higher in Khoja's system results (13.7%) than in our root extraction system (5.2%). That is due to extracting roots of Arabicized and Proper noun words as well as failing to extract the correct roots for many tri-literal weak and hamzated roots. Such as extracting the root (ربي) for the word (روايات) where the correct root is (روي), and the root (خرر) for the word (أخر), where the correct root is (أخر). Overall, our system extracted more accurate roots (65%) than Khoja's system (59%) with an improvement rate of 6%.

Figure 3. Proposed Algorithm vs Khoja's Testing Result

TABLE VII. PROPOSED ALGORITHM VS KHOJA'S TESTING RESULTS

	Proposed Algorithm		Khoja's Algorithm	
# Total Roots Extracted	3061	70.5%	3162	72.8%
# Inaccurate Extracted Roots	224	5.2%	593	13.7%
# Accurate Extracted Roots	2837	65.4%	2569	59.2%

Number of total words processed is equal to 4341

VI. CONCLUSION

In this paper we presented an improved root extraction algorithm for Arabic words, which is based on morphological analysis and linguistic constraints. The algorithm handles the problems of infixes removal by eliminating prefixes, suffixes while checking the word against a predefined list of patterns. Also, the problem of extracting the roots of weak, hamzated, eliminated-long-vowel and tow-letter geminated words has been handled by identifying linguistic based rules to replace, eliminate or duplicate certain letters where needed. The experiments and testing were conducted by using thousands of Arabic words gathered from an online Arabic corpus which is collected to aid Arabic language based research [15]. Human judgment was applied to evaluate the results and accuracy of the algorithm. The algorithm is introduced with the aim of supporting Arabic stemming/root extracting tools. The results obtained shows that our proposed root extraction algorithm is promising and is worth being applied in various Arabic language processing programs.

REFERENCES

- [1] M. K. Saad, W. Ashour, "Arabic morphological tools for text mining," Proc. of the 6th IEEE International Conference on Electrical and Computer Systems, Nov. 2010.
- [2] S. Ghwanmeh, R. Al-Shalabi, G. Kanaan, S. Rabab'ah, "Enhanced algorithm for extracting the root of Arabic words," Proc. of the 6th IEEE International Conference Computer Graphics and Visualization, 2009, pp.388-391.
- [3] L. S. Larkey, L. Ballesteros, M. E. Connell, "Improving stemming for Arabic Information Retrieval: light stemming and co-occurrence analysis," Proc. of the 25th SIGIR, Tampere, Finland, 2002, pp. 275-282.
- [4] J. A. Haywood and H. M. Nahmad. A new Arabic grammar of written language. London: Lund Humphries Publishers. 1998.
- [5] A. Alzamil, "The relationship between roots and patterns: new classification for Arabic language roots (العلاقة الصرفية بين الجذور)", In proceedings of the Int. Symposium on Computers & Arabic Language Riyadh, KSA. 2007.
- [6] R. Alshalabi, "Pattern-based stemmer for finding Arabic roots," Information Technology Journal, vol. 4, no. I, 2005, pp. 38- 43.
- [7] Y. Kadri, J.Y. Nie, "Effective stemming for Arabic Information Retrieval," International conference at the British Computer Society, London, 23 Oct. 2006, pp.68-74.
- [8] G. Kanaan, R. Al-Shalabi, M. Ababneh, A. Al-Nobani, "Building an effective rule-based light stemmer for Arabic language to improve search effectiveness," International Conference on Innovations in Information Technology, IIT 2008, Dec. 2008, pp 312-316.
- [9] S. Khoja, and R. Garside. 'Stemming Arabic text', Computer Science Department, Lancaster University, Lancaster, UK, 1999.
- [10] R. Al-Shalabi, G. Kanaan and H. Muaidi, "New approach for extracting Arabic roots," Proceedings of the International Arab Conference on Information Technology. Alexandria, Egypt, 2003, pp. 123- 131.
- [11] H. Al-Serhan, A. Ayeshe, "A trilateral word roots extraction using Neural Network for Arabic," The 2006 International Conference on Computer Engineering and Systems, Nov. 2006, pp. 436-440.
- [12] M. Sawalha, M. E. Atwell. "Comparative evaluation of Arabic language morphological analyzers and stemmers," Proceedings of COLING 2008 22nd Int. Conf. on Computational Linguistics, 2008.
- [13] S. Eldin, Development of a computer-based Arabic lexicon. In proceedings of the Int. Symposium on Computers & Arabic Language (ISCAL) Riyadh, KSA. 2007.
- [14] A. Al-Rajhi. التطبيق الصرفي. Dar Alnahdhah Alarabiyah Publishers. 2004.
- [15] L. Al-Sulaiti. Latifa Al-Sulaiti's Website: Arabic Online Corpus. 2009. Retrieved from: <http://www.comp.leeds.ac.uk/eric/latifa/research.htm>