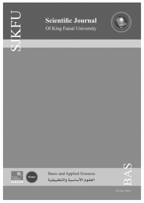
Scientific Journal of King Faisal University:
Basic and Applied Sciences



An Algorithm to Analyze Arabic Verbs Morphologically

Nimet Hamid Kassad [1] and Muhammad Rabee Hatim Shaheen [2]

[1] Department of Arabic language, Faculty of Arts and Humanities, Al-Baath University, Homs, Syria
[2] Department of Informatics Engineering, Faculty of Informatics Engineering, Al-Baath University, Homs, Syria

LINK	RECEIVED	ACCEPTED	PUBLISHED ONLINE	ASSIGNED TO AN ISSUE
https://doi.org/10.37575/b/cmp/230061	31/10/2023	08/02/2024	08/02/2024	01/06/2024

NO. OF WORDS	NO. OF PAGES	YEAR	VOLUME	ISSUE
3172	4	2024	25	1

Abstract

Morphological analysis describes a word in isolation from the context, which is an important stage when trying to automatically understand and interpret the text. Many studies have provided morphological analyzers of Arabic verbs. However, although accurate, they do not deal with all cases of Arabic verbs. This research introduces an algorithm to analyze Arabic verbs morphologically based on regular expressions then on Morphological Database. The advantages of this algorithm are that it deals with verbs in their morphological cases, extracting their possible structures and some of their morphological data. This algorithm yielded highly accurate results after being tested on a sample of vocabulary and text. Furthermore, the extracted data is completely error-free.

KEYWORDS
language processing, match, pattern, regular expression, root, word structure

CITATION
Kassad, N.H. and Shaheen, M.R.H. (2024). An algorithm to analyze Arabic verbs morphologically. *Scientific Journal of King Faisal University: Basic and Applied Sciences*, **25**(1), 41–4. DOI: 10.37575/b/cmp/230061

1. Introduction

Natural language processing is a science that combines linguistics and computer science. Linguistics provides a description of the linguistic material, while computer science automates this description and produces programs and tools such as machine translation and text proofreading.

This paper considers how to process the Arabic language automatically. It answers the question of how the roots of Arabic verbs can be extracted and how their patterns, possible tenses, and parts can be automatically recognized.

Previous studies have presented tools and algorithms to process Arabic verbs automatically (see Section 2). However, the mistake rate in their outputs was inconsistent, and none of them were error-free. Furthermore, some tools only dealt with certain cases of the verb, such as triple verbs.

The difficulty of processing Arabic verbs is due to two features of the verb structure. The first one is diversity; the verb structure is consisted of triple roots, such as “درس,” or quadruple roots, such as “دحرج”. A morphological phenomenon called “derivation” also creates a new verb structure by adding certain characters to the root, such as deriving the verb “قاتل” from “قتل” This phenomenon leads to multiple verb structures. Therefore, to know the letters of the root, we must first know what letters have been added to the root.

The second one is the changes in verb structure; sometimes, changes occur for morphological reasons, and as a result, the verb changes its form. The change may affect the original verbs or their derivatives, as shown in Table 1.

Table 1: Examples of verb structure change

Type Change	Verb	Original Verb	Pattern
Draw "Hamza"	قرئ	قرأ	فعل
Replace vowel	يقول	يقال	يفعل
Remove	عد	عاد	فل
Replace + diphthong	ازدهر	ازتهر	افتعل
Remove + diphthong	كنّا	كننا	فلنا
Remove "Hamza"	يرى	يرأى	يفل

The verb pattern also follows the verb structure, so if the root letters are unknown, the verb pattern will not be known.

The paper follows the following order: Section 2 presents the relevant works, Section 3 explains the working of the proposed algorithm, Section 4 presents the testing of the proposed algorithm and its results, and Section 5 reports the results and looks to future work in this area.

2. Related Works

Many previous studies have presented proposals describing how to analyze Arabic verbs automatically, but they have mostly focused on extracting roots and patterns. Reference is made to several such studies that have seen positive results, evaluating their method and limitations.

- Yousfi (2010) presented a morphological analysis of Arabic verbs using surface patterns linked to a database of surface patterns.

  Limitations of this study: Many errors occur in words not associated with surface patterns.
- Abuata *et al.*, (2011) presented an algorithm to extract the roots of Arabic words. This algorithm removes valid affixes in an input word according to a set of rules. It then generates possible roots for the word and checks them in the root dictionary. Finally, it generates patterns for the word by using a root from each of the possible roots and checks them against a set of valid Arabic patterns.

  Limitations of this study: Mistakes that occur due to:
  - Spelling such as "وبارك"→"ارك"
  - Over-stemming "ليبلو"→ "بل"
  - Other issues "ووقاهم"→"قا"
- Yaseen and Hmeidi (2014) proposed an algorithm called the word substring stemming algorithm (www-based), which is used to extract the roots of Arabic words and their patterns. The algorithm extracts the substring from the word to check it against the root file and generates patterns that allow the input word to check for patterns by using the root’s (التصاريف) file.

  Limitations of this study: It does not deal with words that do not contain all letters of the root, such as “مدّ.” Additionally, it does not deal with one-letter words such as “ع.”
- Al-Kabi *et al.* (2015) presented an algorithm to extract the roots of Arabic words; however, it only works on trilateral roots. This method depends on matching the stemmed word, with all the patterns having the same word number as the stemmed word. Then, it removes the same letters except the main letters to return to the root.

  Limitations of this study: It extracts the roots of Arabic words, but only works on trilateral roots. Incorrect roots are produced either by over- or under-stemming. It also cannot deal with words that are less than

Corresponding Author: Nimet Hamid Kassad 00963959815062, nimatkassad@gmail.com

four letters, and It cannot process short words where vowels are removed.

- Thalji *et al.*, (2018) presented an algorithm to extract the Arabic roots according to set rules. They classified Arabic letters into two main groups: consonant letters that belong to the root and non-consonant letters that sometimes belong to the root and sometimes do not.

  Limitations of this study: The algorithm cannot find the root of words containing three consonant letters where some of them do not belong to the root. It also cannot deal with replacing a consonant with a vowel, because this rarely happens.
- Azman (2019) presented a tool that extracted the roots of Arabic verbs, called "RootIT". This tool requires the use of bottom-up and top-of-bottom analysis together.

  Limitations of this study: The tool cannot deal with verbs that have a pronoun attached by a suffix, such as "يستسقوهم." Also, it does not replace roots that have changed form during the morphology process, such as: "يتراءى" ⟶ "رءى". Hamza should be changed to "أ".
- Othman *et al.*, (2020) presented an algorithm made up of two levels of analysis: Sentence-level analysis, which uses an Arabic context-free grammar parser to identify the sentence and its parts, and lexical-level analysis, which extracts the verb root. The regular expression engine matches the input word with all its prefixes, derivatives, and suffixes to extract the root from the derivative. The hash function then generates an index for each root and Hash table to check whether the inserted derivative is a verb.

  Limitations of this study: The algorithm deals only with trilateral roots. It does not deal with verbs that contain vowel letters. It does not deal with trilateral verbs that contain double letters, and it does not deal with "افتعل" when the "ت" is replaced with "د" "ط" or "ظ".
- Khafajeh *et al.*, (2021) proposed an algorithm to extract roots from Arabic words based on a modified successor variety algorithm. They developed the body of the proposed text in such a way that it contains multiple morphological forms for the word to be rooted.

  Limitations of this study: There is a deficiency in how this algorithm deals with vowels.

3. Proposed Algorithm

The algorithm uses regular expressions and Morphological Database to analyze verbs. The regular expressions are built on the structure of Arabic verb roots and their morphological changes and derivations. Morphological Database contains morphological data for the verb structure, as shown in Figure 2.

Figure 6 shows the flowchart for the proposed algorithm, which consists of the following steps:

Step 1:

The regular expression matches the input word; it consists of [prefixes={$Pr_1, \ldots, Pr_n$} + substring verb + suffixes = {$S_1, \ldots, S_n$}]. Prefixes and suffixes are letters that touch the verb and are not a part of the root. A substring verb starts and ends with the letter of root, it contains the root letters and sometimes an infix (a derived letter). An abstract verb consists of a substring verb – infix; it contains only root letters. Step 1 aims to identify the structure of the word and retain its parts; Figure 1 shows an example.

Figure 1: Identification of the structure of the verb "يتسابقون"

يتسابقون					
ـون S_1	بقـ Root letter	ـا infix	ـسـ Root letter	ـتـ Pr_5	يـ Pr_4
Suffixes	Substring verb			Prefixes	

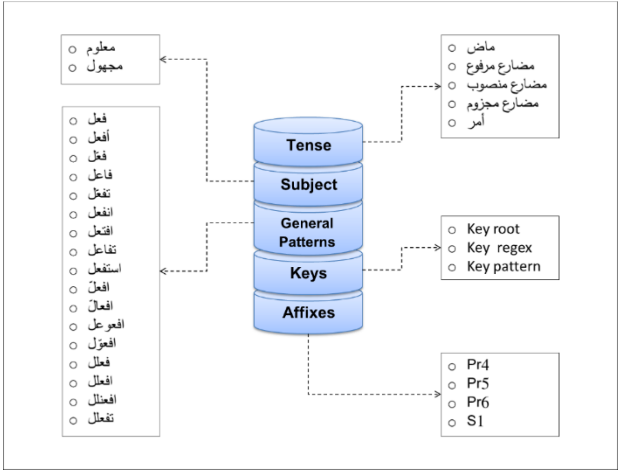
Step 2:

The algorithm checks the structure of the abstract verb using a set of regular expressions containing all the abstract verb structures; each verb has a unique key, which is kept when the match is successful. Here, the letter type and its shape are important for matching purposes, as the regular expressions do not include "[ء-ي]" but rather "[ب-هـ]".

Step 3:

The algorithm calls possible morphological data from Morphological Database; the calling is based on the key that was retained in Step 2 and a set of rules, such as a rule regarding tense. For example, if the prefix contains "ل," the algorithm just chooses the appropriate tense {"مضارع منصوب" ، "مضارع مجزوم"}.

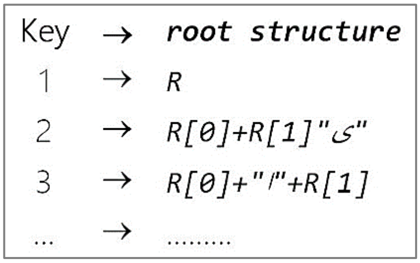
Figure 2: Morphological Database



Step 4:

The algorithm modifies the abstract verb to generate the root; for example, by deleting a letter that appears more than once, consecutively. The list of root structures contains 59 structures, covering all root forms; however, the algorithm only uses the root structures that match the root key. Figure 3 shows the root structures for Keys 1, 2, and 3.

Figure 3: List of root structures

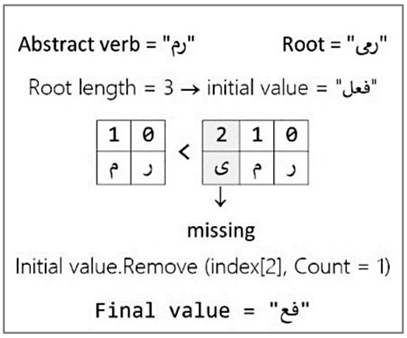


Key	→	root structure
1	→	R
2	→	R[0]+R[1]"ى"
3	→	R[0]+"ا"+R[1]
...	→	....

Step 5:

The algorithm generates an abstract verb pattern. It gives an initial value to this abstract verb pattern according to the root length; if the root length is two or three, then the initial value is "فعل"; if it is four, then the initial value is "فعلل." The algorithm then compares the lengths of the abstract verb and root; if their lengths are equal, the final value is the initial value. If the abstract verb is smaller than the root, the algorithm matches them, and if it does not find one of the letters, the algorithm retains the index of the missing letter and removes a letter from the initial value based on this index. Figure 4 shows how the abstract verb pattern of the verb "ارم" can be generated.

Figure 4: Generating an abstract verb pattern


Abstract verb = "رم"
Root = "رمى"
Root length = 3 → initial value = "فعل"
1 0
م ر
<
2 1 0
ى م ر
missing
Initial value.Remove (index[2], Count = 1)
Final value = "فع"

Step 6:

The verb pattern consists of the word parts generated in Step 1 (prefixes and infix) and the abstract verb pattern generated in Step 4. The list of verb patterns contains 11 structures, each with a specific order of pattern parts. Figure 5 shows examples of this list. The algorithm chooses the verb structure according to key patterns.

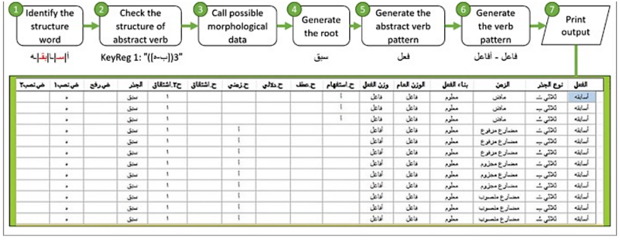
Figure 5: List of verb patterns

key	→	verb pattern
1	→	$Pr_4 + pr_5 + absVP$
2	→	$Pr_4 + pr_5 + absVP[0] + infix + SubString.absVP[1]$
…	→	………

Step 7:

The algorithm prints the outputs. Figure 6 shows how the outputs of the proposed algorithm for the verb "أسابقه" are printed within the interfaces.

Figure 6: Proposed algorithm with example input: "أسابقه"


1 Identify the structure word
2 Check the structure of abstract verb
3 Call possible morphological data
4 Generate the root
5 Generate the abstract verb pattern
6 Generate the verb pattern
7 Print output
أ|سـ|ا|بقـ|ـه
KeyReg 1: "([ـهـب])3"
سبق
فعل
فاعل - أفاعل

4. Accuracy

The proposed algorithm test is based on two basic samples:

A. Vocabulary: this contains 5,798 unique verbs. It uses the variety in the verb models to ensure a comprehensive test. Table 2 shows this diversity by displaying the number of sample verbs. The source of these models are the root models; there are 42 in total, 13 for the consonant root and 29 for the vowel root. Many sample test verbs alternate with 27 prefixes and 21 suffixes.

Table 2: The number of sample verbs

Verb Tense	Object	Consonant Verbs	Vowel Verbs	Total
ماض	معلوم	156	296	452
	مجهول	161	291	452
مضارع مرفوع	معلوم	177	307	484
	مجهول	328	988	1316
مضارع منصوب	معلوم	161	291	452
	مجهول	312	972	1284
مضارع مجزوم	معلوم	167	301	468
	مجهول	159	289	448
أمر	معلوم	155	287	442
		1776	4022	5798

The test results show that the proposed algorithm can capture all verbs in the sample and extract their roots and patterns. Table 3 includes the output of the proposed algorithm after analyzing some of the verbs.

Table 3: Output of the proposed algorithm for some of the sample verbs

Input	Prefixes	Possible Roots	Suffixes	Patterns	Tense
بت	-	بات	-	فل	أمر
					ماض
	-	بت	-	فعل	ماض,أمر
	-	بتي, بتى, بتو,بتا	-	فعّ	أمر
	-	وبت	-	عل	أمر
استمع	ست,ا	ماع	-	استفل	أمر
	ا	ستمع	-	افعللّ	ماض,أمر
	ست,ا	مع	-	استفعل	ماض,أمر
	ت,ا	سمع	-	افتعل	ماض,أمر
	ست,ا	معي, معى ,معو ,معا	-	استفع	أمر
يك	ي	وكي,وكى	-	يع	مضارع مجزوم
		وك	-	يعل	مضارع مجزوم, مضارع منصوب
		كان	-	يف	مضارع مجزوم
يحرنجمون	ي	حرجم	ون	يفعنلل	مضارع مرفوع

B. Variety of texts: it aims to validate regular expressions so that it can recognize and select words without mistakes. The results show that the proposed algorithm can do this without making any mistakes, giving all the morphological probabilities corresponding to the verb. In addition, the proposed algorithm captures all the words that could be a verb, whether the word in the text is a verb or not. It does this by analyzing the text morphologically, isolated from the context.

In addition to the above, this algorithm has been tested on verbs that other algorithms have failed to analyze. Tables 4 and 5 show the analysis of these verbs by the proposed algorithm and by other systems.

Table 4: A comparison of the proposed algorithm's results with those of other systems (by root)

Verb Tested	Proposed Algorithm	Other Systems	
	Root	System	Root
وبارك	بار، بر، برا، برك، برو، بري، برى	Abuata *et a*l., 2011	ارك
ليبلو	بلا، بلو	Abuata *et al.*, 2011	بل
ووقاهم	وق، وقي، وقى	Abuata *et al.*, 2011	قا
فتبينوا	بان، بين	Abuata *et al.*, 2011	تبي
مد	ماد، ومد، مدى، مدي، مدو، مدا، مد	Yaseen and Hmeidi, 2014	-
ع	وعى، وعي	Yaseen and Hmeidi, 2014	-
تراءى	رأى	Azman, 2019	رءي
لاذ	لذ، لاذ، لذا، لذي، لذى، لذو	Azman, 2019	لذذ

Table 5: A comparison of the proposed algorithm's results with those of other systems (by root and pattern)

Verb Test	Proposed Algorithm		Other Systems		
	Root	Pattern	System	Root	Pattern
دق	ودق، دقى، دقي، دقا، دقو، دق	عل، فعّ، فعل، فل	Othman *et al.*, 2020	دقق	فعَّ
جلا	وجل، جلا، جل	عل، فعل	Othman *et al.*, 2020	جلو	فعا
ادعى	دعي، دعى، دعو، دعا	تفعّل، افتعل	Othman *et al.*, 2020	دعو	افدعل
اذرع	ذرع	تفعّل، افعل، افتعل، افعلّ	Othman *et al.*, 2020	ذرع	افدعل
ازدجر	زجر	افتعل، تفعلل، افعللّ	Othman *et al.*, 2020	زجر	افطعل
التقى	لقى، لقي، لقو، لقا	افتعل	Thalji *et al.*, 2018	لتق، لقى، تقى	الفعل، افتعل، افعلى

Tables 4 and 5 indicate that there were errors in the outputs of some systems, while the proposed algorithm was able to extract the possible roots and patterns in Table 5. There may be roots in the set of possible roots that do not belong to the dictionary; this is because the proposed algorithm is not associated with a dictionary. For example, the proposed algorithm extracts two possible roots for the verb"اضطلعوا" : ("ضلع", "ضطلع"). Although the root "ضطلع" is a quadrilateral root according to its verb structure, it does not belong to the dictionary.

Conclusion

This paper has presented a proposed algorithm for the analysis of Arabic verbs using regular expressions and Morphological Database. The test results are encouraging, as the proposed algorithm was able to extract the verb roots and their patterns and provide error-free morphological data. It also outperformed other algorithms' verb analysis, because it solved the problems faced by other algorithms, such as how to deal with verb vowels and instances where the verb contains a repeated character.

In the future, a digital dictionary will be created that contains the roots of Arabic verbs and their derivational patterns according to a numerical indicator. The dictionary will then be linked with the proposed algorithm to remove the possibility of errors in the algorithm's output.

Biographies

Prof. Muhammad Rabee Hatim Shaheen

Department of Informatics Engineering, Faculty of Informatics Engineering, Al-Baath University, Homs, Syria, 00963990482650, rabee.shaheen@uok.edu.sy

Shaheen, from Syria, has a Ph.D. in software engineering from the University of Grenoble (France). He is a professor at the University of Kalamoon. He has worked as a Vice Dean of the Faculty of Informatics Engineering and Head of the Information Technology Department. He has published research in peer-reviewed scientific journals in English and French, with a focus on software testing and management. He wrote the book *Visual Basic.net* and has co-authored a Java series on object-oriented programming and building applications (visual interfaces and databases).
ORCID ID: 0009-0009-1748-1035.

Nimet Hamid Kassad

Department of the Arabic Language, Faculty of Arts and Humanities, Al-Baath University, Homs, Syria, 00 963 95 9815062, nimatkassad@gmail.com

Kassad, a Palestinian Syrian, has a master's degree in teaching the Arabic language from the Higher Institute of Languages at Al-Baath University (Syria). She was a lecturer at the College of Education and a proofreader and office worker at Aphamea Translation and Language Services. She is an Arabic language teacher for non-native speakers, working with children and adults (online). She carries out research and projects in Arabic computer linguistics and develops language games, Arabic language curricula, and interactive lessons for various levels and age stages.
ORCID ID: 0009-0007-8809-2948

References

Al-Kabi, M.N., Kazakzeh, S.A., Ata, B.M.A., Al-Rababah, S.A. and Alsmadi, I.M. (2015). A novel root based Arabic stemmer. *Journal of King Saud University-Computer and Information Sciences*, **27**(2), 94–103. DOI: 10.1016/j.jksuci.2014.04.001

Azman, B. (2019). Root identification tool for Arabic verbs. *IEEE Access*, **7**(n/a), 45866–71. DOI: 10.1109/ACCESS.2020.2999259

Othman, M.T.B., Al-Hagery, M.A. and El Hashemi, Y.M. (2020). Arabic text processing model: Verbs roots and conjugation automation. *IEEE Access*, **8**(n/a), 103913–23. DOI: 10.1109/ACCESS.2020.2999259

Khafajeh, H.H., Yousef, N.A. and Al-Tarawneh, H. (2021). Arabic Words Root Extraction Using Modified Successor Variety. In: *2021 22nd International Arab Conference on Information Technology (ACIT)*, Muscat, Oman, 21–23/12/2021. DOI: 10.19101/IJACR.2017.733023

Abuata, B., Sembok, T. and Bakar, Z. (2011). A Rule-based arabic stemming algorithm. In: *Proceedings of the European Computing Conference, ECC* (Vol. 11), Paris, France, 28–30/04/ 2011.

Thalji, N., Hanin, N.A., Hani, W.B., Al-Hakeem, S. and Thalji, Z. (2018). A novel rule-based root extraction algorithm for Arabic language. *International Journal of Advanced Computer Science and Applications*, **9**(10), 120–8. DOI: 10.14569/IJACSA.2018.091015

Yaseen, Q. and Hmeidi, I. (2014). Extracting the roots of Arabic words without removing affixes. *Journal of Information Science*, **40**(3), 376–85. DOI: 10.1177/0165551514526

Yousfi, A. (2010). The morphological analysis of Arabic verbs by using the surface patterns. *IJCSI International Journal of Computer Science Issues*, **7**(3), 33–6.