Hume Studies,
ISSN / eISSN 0319-7336 / 1947-9921
Publisher HUME SOCIETY, OCCIDENTAL COLLEGE, DEPT PHILOSOPHY,LOS ANGELES, USA, CA, 90041
Volume 26-Issue 1-(2026)
DOI:https://doi-0004.org/6812/17869516576382

Morphological Analyzers and Their Role in the Computerization of Arabic Morphology: The Qutrub Program as a Case Study

AIAD Salima

Center for Scientific and Technical Research on Arabic Language Development (CRSTDLA), Ouargla Unit (Algeria)

Email: **aiad.s70@gmail.com**

Received: 11/05/2026 ; Published: 17/08/2026

Abstract

Natural language processing is a product of the cross-fertilization between linguistics and computer science; it refers to the naturalization of technical knowledge mechanisms within the computer in order to facilitate the input and retrieval of knowledge. Specialists in computational linguistics have taken an interest in designing programs dedicated to analyzing the phonological, morphological, syntactic, and semantic levels of the Arabic language, in an effort to serve it and to allow it to benefit from what modern technology offers, so that it may rise to the rank of the world's major languages. This paper is devoted to discussing the automatic processing of the morphological level of Arabic, that is, morphology, which is concerned with derivation and inflection. The word is the central object of study at this level, since research revolves around its origin, its form, its pattern (wazn), and the identification of which of its sounds are radical and which are augmentative, among other matters. The discussion of the automatic processing of the morphological level in Arabic will proceed through the presentation of a model morphological analyzer, namely the Qutrub analyzer, in order to identify its characteristics, how it deals with the Arabic language, and what it offers in this field.

Keywords: automated processing; computational linguistics; Arabic language; morphological level; Qatrab software.

Introduction

The world today is witnessing rapid technological development that has brought about a genuine revolution in the relationship between human beings and the universe. The human mind has managed to invent the computer and to equip it with a range of systems and programs that simulate human intelligence, making this machine pliable and responsive to human command, serving and meeting human needs. Among the most prominent fields from which humans have benefited in this regard is the field of computational linguistics.

Computational linguistics is an interdisciplinary field between linguistics and computer science concerned with studying language through computer programs. Mazen al-Wa'r defines it as the scientific framework that investigates human language as a natural tool to be processed by the machine, and states that the principles of this discipline are composed of general linguistics with all its analytical levels, namely the phonological, syntactic, and semantic, together with electronic computer science, artificial intelligence and logic, and mathematics, all combining to form the principles of computational linguistics[1].

The Arabic language is characterized by features that differ from those of other languages, and since computer systems were originally built to suit the nature and characteristics of Western languages, chief among them English, anyone examining the situation of Arabic from a computational standpoint finds that it faces numerous obstacles hindering its computerization. Adapting computer systems to accommodate the features of Arabic therefore requires sustained effort on the part of both computer scientists and linguists, a burden that computational linguistics continues to bear today. At the same time, despite these difficulties and obstacles, Arabic possesses internal characteristics and inherent scientific dispositions that make it well suited to entering the world of computing, since there are numerous points of similarity and compatibility between the features of Arabic and computer systems. Arabic is a mathematical, derivational language governed by precise algorithms; moreover, it is characterized by the root system, morphological patterns (awzan), rhythm, and derivation, among other features that may be distinctive to it.

In an effort to identify the position of the Arabic language in the world of computing and the extent to which it has benefited from its applications, this article, titled “Morphological Analyzers and Their Role in the Computerization of Arabic Morphology: The Qutrub Program as a Case Study,” aims to shed light on the morphological level of Arabic, a cornerstone of its linguistic system, and on what computational linguistics has offered this level, and how it is processed automatically, through an examination of one morphological analyzer, namely the Qutrub analyzer. We approach this topic by way of the following research question: What is meant by morphological analyzers designed to process the Arabic language, and does the Qutrub analyzer represent a model suited to the Arabic morphological system? Does it offer solutions to the problems that hinder the computerization of this level? To study this topic, we have adopted the descriptive method.

Natural Language Processing

Natural language processing is a subfield associated with artificial intelligence and computational linguistics, concerned with the problems of automatic understanding and generation of natural languages. This field aims to convert human linguistic samples and models into formal representations at the level of automatic understanding, while at the level of generation the goal is to convert data stored in computer databases into human linguistic samples.

The term natural language processing is sometimes used interchangeably with other terms such as language engineering and computational linguistics. All these terms converge on a single aim: adapting natural language, with all its complexities, to the binary logic of zero and one within computer programs and systems.

Natural language processing thus represents a point of intersection between two fields, linguistics and computer science, aimed at serving natural language and facilitating engagement with it. Computer specialists study the foundations of natural languages in order to develop high-level programming languages, while linguists supply the characteristics of natural languages and the problems they raise across various domains to computer specialists, who in turn build linguistic models and analyze different branches such as computational phonetics, computational morphology, computational syntax, and computational lexicography.

It can therefore be said that natural language processing aims to remove the barriers between the computer and natural languages, adapting human language to computer programs that are highly efficient in storing and retrieving information and processing data quickly and accurately. The applications of natural language [2]processing have multiplied to include machine translation, automatic summarization, automatic language generation, information extraction and retrieval, text-to-speech conversion, speech understanding, spelling and grammar checking, linguistic statistics, teaching Arabic to non-native speakers, and automatic morphological analysis, among others. In this paper we present one of these applications, namely the automatic morphological analysis of Arabic.

The Morphological System of the Arabic Language

Like other Semitic languages, the morphological system occupies a distinguished place within the overall linguistic system of Arabic, forming, in effect, the connecting knot of the elements of the linguistic system: it is the foundation of phonology, the gateway to syntax, the basis for organizing the lexicon, and, beyond that, the hot frontline where the forms and meanings of the language meet[3].

The morphological system of Arabic qualifies it to stand at the forefront of world languages in the field of computing, since it is a derivational language that builds its vocabulary on the root and the pattern (wazn), two mathematical components that many other languages lack. The root is the origin of the word, providing its basic structure, while the pattern provides its overall framework.

Challenges Facing the Computerization of Arabic Morphology

The Arabic language faces a set of challenges that stand as obstacles to its computerization, the most important of which are the following:

Context (al-siyaq / al-maqam)

Specialists in computational linguistics have built a range of computer systems consisting of a logical software architecture grounded in strict algorithms that do not operate on conjecture or probability; the rules formulated for this purpose must therefore be formal and decisive, admitting no more than one interpretation for a given case[4]. This creates a serious problem for Arabic, a language of wide-ranging semantic possibilities, since a word in Arabic carries multiple meanings depending on the context in which it occurs, which makes it difficult for the software system to determine the intended meaning. This problem does not arise in English, the language of the producer that designed these programs according to the features of its own language, and since Arabic occupies the position of the consumer, it remains at the mercy of a producer that does not take the features of its linguistic system into account.

Description versus Formalized Specification (al-wasf wa al-tawsif)

Nihad al-Moussa drew a distinction between description and formalized specification when discussing the problems facing the computerization of language, terming what is produced for human beings "description" and what is produced for the computer "formalized specification." He based this distinction on the traditional Arab description of language, which relied heavily on the receiver, who contributes his intuition, natural linguistic sense, and experience to understanding what is addressed to him, whereas what is directed at the machine is a formalized specification, because the machine lacks these human faculties. He illustrated this with the sentence “Moussa entered the hospital” (dakhala Moussa al-mustashfa), which a human being understands in that word order (verb + subject + object) and understands equally well if the order is reversed to “the hospital entered Moussa” (dakhala al-mustashfa Moussa), i.e., verb + object + subject. A human being, through intuition and experience, is not confused by the two phrasings and recognizes that Moussa is the subject regardless of whether it comes first or last in the sentence, whereas the machine needs a list of lexical data about each word specifying its behavior in the sentence before it can grasp that the hospital cannot enter Moussa. He gave a further example with two phrases, “the horse saw the whip” (ra'a al-hisan al-sawt) and “the horse heard the sound” (sami'a al-hisan al-sawt), noting that a human being hearing the two sentences recognizes the difference between the words “whip” (sawt) and “sound” (sawt), even though they sound alike, whereas the computer needs an elaborate set of functions to establish the distinction and grasp the meaning intended in each sentence[5].

Adequacy (al-kifaya)

Linguistic adequacy, or competence, is a relatively new term in linguistic discourse, though old in its underlying meaning; Ibn Khaldun referred to it under the notion of linguistic aptitude (al-malaka al-lughawiyya), describing it as a firmly rooted faculty that develops in the user of a language after prolonged exposure to hearing it, followed by proper instruction and repeated mental rehearsal through images and depictions until it becomes firmly established. Chomsky later addressed this faculty under the term linguistic competence, defining it as the knowledge possessed by the ideal speaker-listener of a language[6].

This competence is what a human being must possess in order to master the language used for communication within society. When this notion of competence is projected onto the computer, matters differ, since in this case it must rest on four pillars:

- First: attempting to input the rules of the Arabic language into the computer system, including its phonological system, its morphological patterns, its syntactic organization, the rules of its grammar (i'rab), the meanings of its words, its various usages, its rhetorical styles, and the rules of its orthography.
- Second: producing an unlimited number of correct linguistic performances on the basis of initial data, which is what we call analogy and representation.

- Third: serving as a reference for distinguishing correct usage from error.
- Fourth: using language in accordance with contexts and requirements.

Script

Arabic script consists of letters and diacritics. The letters make up the Arabic alphabet, comprising twenty-eight letters written from right to left, while the diacritics include the damma, fatha, and kasra, in addition to the shadda (gemination mark), the madd (elongation mark), and tanwin (nunation)[7]. Because of these features, which distinguish Arabic script and its writing system from other languages, the Arabic script has faced a number of computational problems, the most important of which are:

- The keyboard, which was designed specifically for the producer's language, English.
- The right-to-left direction of the script, contrary to English.
- The multiple forms a single Arabic letter can take depending on its position within a word, such as the hamza, the kaf, and the ha, among others.
- The joining and connection of letters to one another, in contrast to the separated letters of English[8].

Automatic Semantic Analysis

Formalizing the semantic system of Arabic represents the nucleus and backbone of automatic processing, since virtually all operations of automatic processing of linguistic structures rest, in one way or another, on it and refer back to it. This is considered by those concerned with computational linguistics to be the most difficult area to handle, given that the meanings of words are self-evidently tied to a depth of understanding that remains beyond the computer's reach. Perhaps the best approach, in seeking to build this semantic system, would be to draw on a corpus of classical Arabic texts, whose material is selected in advance according to scientific criteria agreed upon by Arabic-language specialists[9].

Accordingly, the automatic processing of Arabic texts requires that the computer possess a vast quantity of the meanings that words can carry, meanings determined by the structure of the word and its position within a given context. Semantics is therefore of great importance in Arabic computational linguistics and constitutes a formidable challenge for the automatic processing of Arabic linguistic texts.

Morphological Analyzers

A morphological analyzer is a computer program that studies the structure of a word and reduces it to its original root. The operation of a morphological analyzer requires a list of all possible prefixes, another of all possible suffixes, a list of all basic patterns (awzan), a database of all triliteral and quadriliteral roots, and a database of all invariable (indeclinable) words.

The morphological analyzer program works by studying the structure of the word in order to identify its morphological category, for example, determining whether it is plural or singular, masculine or feminine in form, or, for verbs, whether it is in the past, present, or imperative form, among other features, as well as identifying the root of the word and the augmentative elements added to that root. This is carried out through two operations[10], one called generation and the other analysis (or conversion).

The Qutrub Program for Verb Conjugation

The program was created by Dr. Taha Zerrouki, who released this analyzer in two forms: the first is a website (https://qutrub.arabeyes.org/), and the second is a desktop application[11].

The Qutrub program conjugates Arabic verbs across the different tenses and moods and attaches them to the personal pronouns. It is distinguished by being open source, offering free use to the user in the form of both a website and a desktop program. Qutrub conjugates Arabic verbs with ease and simplicity, helps the user learn and correct their knowledge, and assists beginning learners of Arabic, whether native speakers or non-native learners, in learning verb conjugation, while also benefiting Arabic-language teachers in preparing their lessons.

It should be noted that Qutrub does not rely on a conjugation database; rather, it conjugates verbs by means of a simple algorithm. It does, however, draw on a database to verify user input, to improve the way it interacts with the user in correcting an entered verb, and to anticipate the errors a user is likely to make and suggest the correct verb forms[12].

How to Use the Qutrub Program

The Qutrub program, like other applications, can be installed on computers or smartphones via Google Play. The user searches for “Qutrub: Verb Conjugation Program” and installs it, after which the following screen appears

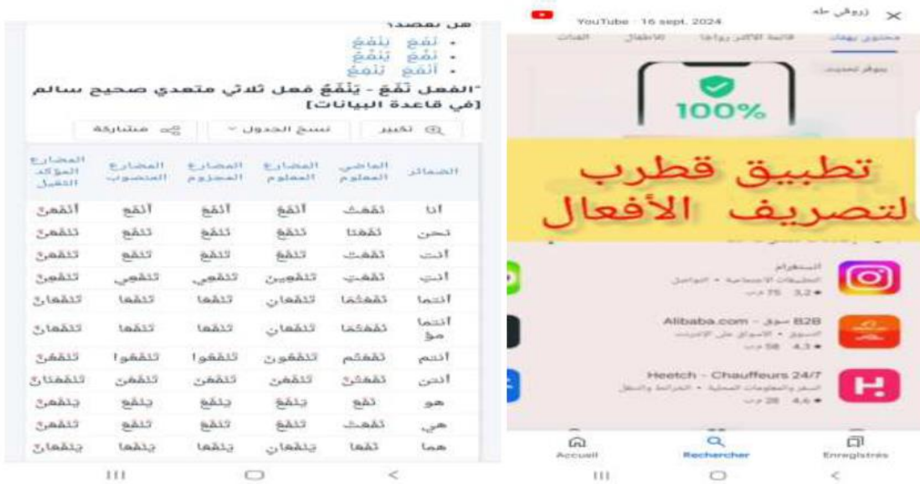
(Figure 1).

Features of the program

The program performs the following functions:

- Conjugates the verb in the emphatic present, the emphatic imperative, and the subjunctive and jussive present.
- Handles hamzation, assimilation (idgham), and vowel-weakening (i'lal).
- Handles weak (irregular) verbs.
- Simplifies the conjugation functions for each tense.
- Improves assistance to the user in conjugating verbs beginning with an elongated alif (alif madd), such as āmana (to believe) and āsa (to console).
- Automatically applies diacritics (tashkil) to non-triliteral verbs when conjugation is requested, so that the user is not obliged to add diacritics to the verb they wish to conjugate.
- Is able to apply diacritics to non-triliteral verbs without relying on a database.
- Includes a list of 17,000 verbs.
- Can conjugate a verb that is not present in the database.

The screen shown in Figure 2 appears at the second step: it presents a model conjugation of the verb qara'a ("to read") across the various tenses (past, present and imperative, indicative present, subjunctive present, indicative-subjunctive present, past passive, etc.). Conjugations of other verbs can also be searched; for example, searching for the conjugation of the verb nafa'a ("to benefit") produces the table shown in Figure 2.

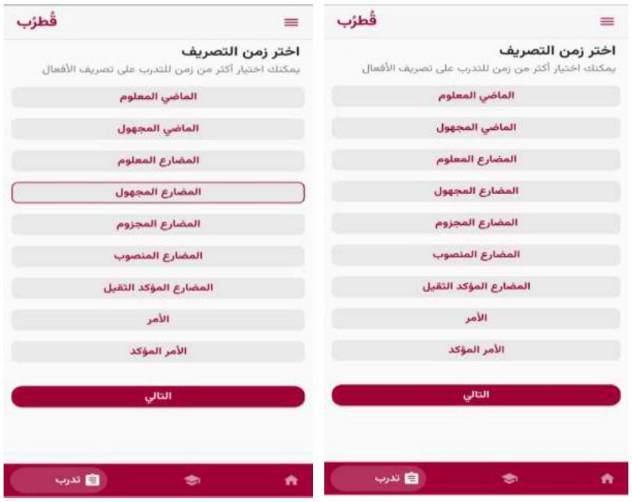


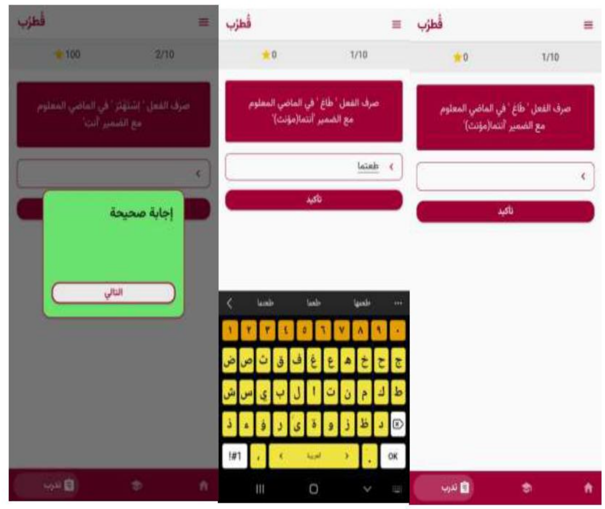
(Figure 2).

The program also offers lessons in Arabic morphology, accessible by moving to the next page and tapping the "Practice" (tadarrab) icon at the bottom of the screen, as shown in Figure 3. These lessons include the classification of the verb according to tense, the accusative, jussive, and indicative moods of the present-tense verb, the classification of verbs into sound-final and weak-final types, the rules governing the construction of the past-tense and imperative verb forms, the accusative case of the present-tense verb governed by an implicit "an," and other lessons dealing with the morphological level of Arabic.

(Figure 3).

The program also offers a language game that allows the learner to practice and train in verb conjugation. The user accesses this by tapping the “Let me practice” (atadarrab) icon at the bottom of the screen, which brings up Figure 4, offering a choice of the available tenses. Having chosen a tense, the next page appears (Figure 5), offering a choice of pronoun: first, second, or third person. Having made this choice, the user taps “Next” to bring up the question screen, also shown in Figure 5.





(Figure 4)

(Figure 5)

The question consists of a verb that the learner is asked to conjugate in the tense and with the pronoun previously selected. Figure 6 shows a sample of the game with a triliteral verb.

(Figure 6).

Figure 7 shows a sample with a quadriliteral verb.

(Figure 7).

Figure 8 shows the evaluation of the answer as either correct or incorrect.

(Figure 8).

Some Criticisms Raised by Users of the Qutrub Program

- With respect to conjugating genuine triliteral verbs that have an established root and conjugation pattern (bab) recorded in the dictionaries, the program performs excellently. However, this is not the true test of the program's capabilities, since a comparable program such as Sarf (a system of derivation and conjugation) conjugates these verbs with the same level of competence, and may even surpass Qutrub in handling irregular verbs.

- The user can disable the triliteral-verb database (Tools → Preferences), in which case conjugation comes to rely on the rules alone, without reference to the database, and the results then differ. For example, when the verb daraba (“to strike”) is conjugated with the database active, three conjugation patterns result, namely daraba/yadribu (kasra), dariba/yadrabu (fatha), and daruba/yadrubu (damma), whereas with the database disabled a single, unusual conjugation results, daraba/yadrubu, formed by disregarding the vocalization of the past tense and arbitrarily choosing the damma for the present tense; tracing further examples makes it difficult to determine the rule governing the choice of pattern.

- When the root-search feature is disabled and the input ‘afw (“pardon”) is entered as a verb, Qutrub conjugates it as huwa ‘afawa to ya‘fu, placing a fatha on the waw of the past-tense form without applying the appropriate vowel-weakening (i’lal).

- Qutrub makes errors in conjugating certain augmented verb forms; for example, when the verb a‘adda (“to prepare,” of the af‘ala pattern) is entered, it conjugates it as a‘dadtu, u‘addu, u‘adid, in the active voice.

- Qutrub neglects to supply diacritics for letters whose vocalization the user has left unmarked, even where that vocalization could reasonably be inferred, as is the case when the user enters the verb istawsa (“to seek advice”).

- The most important virtue of Qutrub is that it draws attention to the need to move beyond a database-oriented way of thinking; however, its designer may have rushed to release it before fully working out the mathematical and non-mathematical rules that generally govern verb conjugation.

- We also observed that the Qutrub program, in its verb-conjugation game, presents unusual and little-used verbs, such as laji-na, iqta‘ada, and zaghara, among other rarely used verbs.

The Response of Taha Zerrouki, Developer of Qutrub

In response to these observations, the developer offered the following remarks. Regarding the claim that a comparable program “may surpass Qutrub in handling irregular verbs,” he stated that he genuinely wished to know which irregular verbs were meant with respect to the choice of triliteral conjugation patterns, noting that the program does provide, through its list of triliteral verbs, three patterns for the verb daraba. As for the missing vocalization of daraba in one of these patterns, he acknowledged this as an error to be corrected. On the point about disabling the “triliteral root” feature, he clarified that there had been a confusion of terms: the program does not rely on triliteral roots but on triliteral verbs, so that an input such as ‘afw becomes a verb, and the resulting conjugation is in that sense correct, since it is not being treated as a root; the program does not rely on roots at all. Regarding the errors in conjugating certain augmented verbs such as a‘adda, ihtalla, istamarra, and istadalla, he confirmed that this was correct and explained that it results from the fact that these verbs' past-tense forms have undergone a substitution through assimilation (idgham); the correct underlying form of istadalla, for instance, should have a sukun on the dal (istadl-la) rather than a fatha, in order for it to be conjugated correctly. He noted that this issue had been addressed in forthcoming beta versions, and that, in fact, this very case led to the discovery of a remarkably consistent and regular underlying behavior of Arabic verbs, a logical regularity that, in versions currently under development, makes it possible to conjugate non-triliteral verbs without requiring the user to vocalize the input at all, yielding results with full diacritics. As for the criticism that Qutrub neglects to supply diacritics the user has left unmarked even when they could be inferred, as with istawsa, he agreed this was entirely accurate, adding that in the new version Qutrub is able to fully vocalize any verb entered: for triliteral verbs it relies on the list of triliteral verbs, while for non-triliteral verbs it is able to infer the vocalization.

Concerning the observation that the program's greatest virtue lies in moving beyond a database-oriented mindset, he agreed that this is indeed the very goal behind Qutrub, though he conceded that its designer may have released it before fully deriving the mathematical and non-mathematical rules governing verb conjugation in general.

He added that the program has, in his view, reached an advanced stage, and that it was released before being fully complete deliberately, in order to gather the opinions of specialists and make it available for testing, since no single person, however capable, can master every detail; the team has learned a great deal from user contributions and expert feedback. He noted that this approach is used in many projects, citing Google's Arabic diacritization program as an example, which was likewise released in an incomplete form in order to benefit from public feedback. He further noted that those interested in exploring some of Qutrub's inner workings could consult the log of verbs entered by visitors to the site.

Conclusion

In concluding this study, we have reached a number of findings, the most important of which are:

- Natural language processing is a subfield associated with artificial intelligence and computational linguistics, concerned with the problems of automatic understanding and generation of natural languages.
- This field aims to convert human linguistic samples and models into formal representations at the level of automatic understanding, while at the level of generation the goal is to convert data stored in computer databases into human linguistic samples.
- The term natural language processing is sometimes used interchangeably with terms such as language engineering and computational linguistics, all of which converge on adapting natural language, with all its complexities, to the binary logic of computer programs and systems.
- Natural language processing represents a point of intersection between linguistics and computer science, aiming to serve natural language and facilitate engagement with it.
- Like other Semitic languages, the morphological system occupies a distinguished place within the overall linguistic system of Arabic, forming the connecting knot of the elements of the linguistic system: the foundation of phonology, the gateway to syntax, the basis for organizing the lexicon, and the frontline where the forms and meanings of the language meet.
- Several problems face the processing of the morphological level of Arabic, the most important being context, adequacy, and the automatic processing of meaning.
- A morphological analyzer is a computer program that studies the structure of a word and reduces it to its original root; this requires a list of all possible prefixes and suffixes, a list of all basic patterns, a database of all triliteral and quadriliteral roots, and a database of all invariable words.
- The Qutrub program conjugates Arabic verbs across the different tenses and attaches them to personal pronouns. It is open source, offering free use as both a website and a desktop program, conjugates verbs with ease, helps users learn and correct their knowledge, assists beginning learners of Arabic, native and non-native alike, and benefits Arabic-language teachers in preparing lessons.

- Users of the program have recorded a number of observations, including: Qutrub makes errors in conjugating certain augmented verbs; for example, entering the verb a‘adda (of the af‘ala pattern) yields a‘dadtu, u‘addu, u‘adid in the active voice.
- Qutrub neglects to supply diacritics for letters the user has left unmarked, even where they could reasonably be modified, as with the verb istawsa.
- We observed that the program's verb-conjugation game presents unusual, little-used verbs such as laji-na, iqta‘ada, and zaghara, among other rarely used verbs.
- We suggest that the program be accompanied by a glossary offering simple

explanations of the verbs used, accessible to students.

---

[1] Mazen al-Wa'r, *Fundamental Issues in Modern Linguistics*, Dar Tlass for Studies, Translation and Publishing, 1st ed., 1988, p. 435.

[2] al-'Arif, 'Abd al-Rahman ibn Hasan, Umm al-Qura University, *The Use of Computational Linguistics in the Service of Arabic Linguistic Studies: Efforts and Results*, Jordan Academy of Arabic Language, 2016, p. 203.

[3] Nabil 'Ali, *Arabic Language and the Computer*, Ta'rib Foundation, Kuwait, 1988, p. 247.

[4]al-Hannash, Mohammed (2002), *Arabic and the Computer (A Quick Reading in Arabic Linguistic Engineering, or an Approach to the Computer Simulation of the Arab Linguistic Brain)*, lecture, Department of Arabic Language and Literature, United Arab Emirates University, p. 2.

[5] al-'Anati, Walid Ahmad, *Arabic Computational Linguistics: A Cultural Perspective*, article available online.

[6] Noam Chomsky, *Aspects of the Theory of Syntax*, Cambridge, MA: MIT Press, trans. fr., Paris: Le Seuil, 1971, p. 12.

[7] Ayadh, Salima, "Automatic Diacritization of Arabic Linguistic Texts between the Specificity of Arabic and the Challenges of Digitization," in *Artificial Intelligence and Its Role in Learning and Teaching Arabic*, refereed scholarly research, ed. Mahmoud Muhammad Qadum, 2024, p. 153.

[8] Mohammed Harrath, "The Computerization of the Arabic Language amid Technical Challenges and Scientific Preparedness," *Al-'Umda Journal of Linguistics and Discourse Analysis*, vol. 0, no. 2, 2022, p. 394.

[9] Wahdan Muhammad Salih Kannali, *Arabic Computational Linguistics: Method and Framework*, paper presented at the Second International Conference on the Arabic Language, Dubai, 30 Jumada al-Akhira 1434 AH, p. 10 ff.

[10] Ma'mun Subhi al-Hattab et al., *The Arabic Script and Technology*, in *Research on the Computerization of Arabic*, King 'Abdullah bin 'Abdulaziz International Center for the Arabic Language, 1st ed., Riyadh, 2015, p. 135.

[11] The application can be downloaded via the following link: https://qutrub.arabeyes.org/download/

[12] See the program overview on its official website: https://qutrub.arabeyes.org/doc/

References

1. al-Hannash, Mohammed (2002). Arabic and the Computer (A Quick Reading in Arabic Linguistic Engineering, or an Approach to the Computer Simulation of the Arab Linguistic Brain). Lecture, Department of Arabic Language and Literature, United Arab Emirates University.

2. Ayadh, Salima. "Automatic Diacritization of Arabic Linguistic Texts between the Specificity of Arabic and the Challenges of Digitization," in Artificial Intelligence and Its Role in Learning and Teaching Arabic, refereed scholarly research, ed. Mahmoud Mohammed Qadum, 2024.

3. al-'Arif, 'Abd al-Rahman ibn Hasan. Umm al-Qura University, The Use of Computational Linguistics in the Service of Arabic Linguistic Studies: Efforts and Results. Jordan Academy of Arabic Language, 2016.

4. al-'Anati, Walid Ahmad. Arabic Computational Linguistics: A Cultural Perspective. Article available online.

5. Mazen al-Wa'r. Fundamental Issues in Modern Linguistics. Dar Tlass for Studies, Translation and Publishing, 1st ed., 1988, p. 435.

6. Mohammed Harrath. "The Computerization of the Arabic Language amid Technical Challenges and Scientific Preparedness," Al-'Umda Journal of Linguistics and Discourse Analysis, vol. 0, no. 2, 2022.

7. Nabil 'Ali. Arabic Language and the Computer. Ta'rib Foundation, Kuwait, 1988.

8. Noam Chomsky. Aspects of the Theory of Syntax. Cambridge, MA: MIT Press. Trans. Fr., Paris: Le Seuil, 1971, p. 12.

9. Wahdan Mohammed Salih Kannali. Arabic Computational Linguistics: Method and Framework. Paper presented at the Second International Conference on the Arabic Language, Dubai, 30 Jumada al-Akhira 1434 AH.
10. Ma'mun Subhi al-Hattab et al. The Arabic Script and Technology, in Research on the Computerization of Arabic. King 'Abdullah bin 'Abdulaziz International Center for the Arabic Language, 1st ed., Riyadh, 2015.
11. Qutrub application download: https://qutrub.arabeyes.org/download/
12. Qutrub documentation: https://qutrub.arabeyes.org/doc/