Journal of King Saud University – Computer and Information Sciences 29 (2017) 141–146

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

AlKhalil Morpho Sys 2: A robust Arabic morpho-syntactic analyzer

Mohamed Boudchiche [a,*], Azzeddine Mazroui [a], Mohamed Ould Abdallahi Ould Bebah [b], Abdelhak Lakhouaja [a], Abderrahim Boudlal [c]

[a] Department of Mathematics and Computer Science, Faculty of Science, Mohammed First University, Oujda, Morocco
[b] Arab Center for Research and Policy Studies, Doha, Qatar
[c] Faculty of Letters and Human Sciences, University Mohammed First, Oujda, Morocco

ARTICLE INFO

Article history:
Received 21 January 2016
Revised 27 May 2016
Accepted 31 May 2016
Available online 6 June 2016

Keywords:
Natural language processing
Analyzer
Lemma
Morphosyntactic parser
AlKhalil Morpho Sys

ABSTRACT

AlKhalil Morpho Sys is a morphosyntactic analyzer of standard Arabic words taken out of context. The system analyzes either partially vowelized words or totally vowelized ones. In this paper, we present the second version of this analyzer. The correction of errors in the database of the first version, and enrichment of this database by missing data allowed us to develop a more accurate version with very high coverage since the percentage of analyzed words exceeds 99%. In addition, we have enriched the morphological features provided by this new version with the lemma tag of the word and its pattern, which are very useful in many applications of Arabic language processing. Furthermore, with the new organization of this database and the improvements brought to its source code, this new version produces very fast analysis.

1. Introduction

With the strong expansion of Arabic texts on the Web, developing tools for Arabic language processing (ALP) has become a necessity. Several researches have been conducted in recent years. These researches concerned both tools such as morphological analyzers and parsers, then applications such as search engines, machine translation, text classification and automatic summarization.

The performance of these applications are partly dependent on the accuracy and efficiency of the tools used in their development. The morphological analyzer is at the top of these tools since several ALP applications use a morphological analysis during the analysis process. Therefore, the development of a morphological analyzer to properly handle all Arabic words and provide maximum morphological information is of great interest for ALP. It should be noted that this system remains a challenge for researchers, and this is particularly due to the richness and complexity of the Arabic language (Sawalha et al., 2013).

In this work, we present Alkhalil2, which is an improved version of Alkhalil Morpho Sys analyzer[1] (Boudlal et al., 2010). This version aims to address the shortcomings of the first version. Indeed, corrections made on its database and its enrichment with missing information has allowed us to develop a more accurate version with very high coverage since the percentage of analyzed words exceeds 99%. In addition, the morphological features provided by this new version are enriched with the lemma of the word and its pattern, which are very useful in many applications of ALP. The new source code and improvements to the structure of the database have greatly increased the speed of analysis. Finally, to make the program easily integrated in other applications, an API version of the code is available.[2]

2. Arabic language characteristics

Arabic is a fusional language where an Arabic word may be a sentence when translated to other languages. For example, the word "سنخبركم" <*snxbrkm*[3]> 'we will inform you' becomes a sentence in English. The Arabic word can be decomposable into

* Corresponding author.
E-mail addresses: moha.boudchiche@gmail.com (M. Boudchiche), azze.mazroui@gmail.com (A. Mazroui).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

[1] https://sourceforge.net/projects/alkhalil/.
[2] http://oujda-nlp-team.net/?p=1299&lang=en.
[3] Buckwalter transliteration.

http://dx.doi.org/10.1016/j.jksuci.2016.05.002
1319-1578/

proclitic, prefix, lemma, suffix and enclitic (Cohen, 1970) (see Fig. 1). Thus, we can have the most complex form of an Arabic word if all these constituents co-occur. The inflected form proclitic + stem + enclitic constitutes the lexical nucleus of the written word. Thus, the word "بمدارسهم" <*bmdArshm*> 'at their schools' has proclitic "ب" <*b*> 'at', the stem "مدارس" <*mdArs*> 'schools' and the enclitic "هم" <*hm*> 'their'.

Clitics (i.e. proclitics and enclitics) are morphemes that convey grammatical information. So, in the written word "بمدارسهم" <*bmdArshm*> 'at their schools', the enclitic "هم" is the construct state. Clitics constitute a finite set, but some combinations can take place between proclitics or enclitics to give an additional list of compound clitics.

Identification of these lexical units (proclitic, stem and enclitic) requires the implementation of methods for selecting the appropriate segmentations of the word among all possible segmentations. However, lack of diacritic marks in written texts makes their analysis complex and ambiguous (Habash et al., 2009). For instance, the non vowelized word "علم" <*Elm*>may be read "عِلْمٌ" <*EilmN*> 'science', "عَلَمٌ" <*EalamN*> 'flag', "عَلِمَ" <*Ealima*> 'he knew' and "عُلِمَ " <*Eulima*> 'It was known'. Thus, an isolated word without diacritic marks can have several interpretations, and its appropriate reading and meaning depend on its context.

To analyze the stem of each potential segmentation of the word, various classifications of the Arabic lexicon, which is estimated at 6×10^{10} distinct words (Darwish and Oard, 2002), can be considered. We adopt a classification based on derived and non-derived words. In adopting this classification, we follow the tradition of Classical Arabic Morphology, according to which derived words are obtained by combining a root and a pattern. The classification is thus orthogonal to the distinction between inflected and derived word forms, which is more common in formal approaches to theoretical morphology.

Derived words are characterized by a root and a pattern. For example, the word "مارسوا" <*mArswA*> 'they practiced' is derived from the root "م ر س" according to the pattern "فاعلوا".

Non-derived class comprises, on the one hand, proper nouns and foreign nouns, and on the other, particles such as determiners (articles), prepositions, adverbs, and conjunctions.

3. Review of the literature

The development of morphological analyzers for Arabic language has aroused the interest of several research teams in recent decades (Al-Sughaiyer and Al-Kharashi, 2004; Farghaly and Shaalan, 2009; Habash et al., 2009; Soudi et al., 2007). The approaches adopted in the development of these analyzers have been conditioned by the fields of application for which these analyzers have been developed. We remind below some systems among the most cited ones in the literature.

- BAMA (Buckwalter, 2002): designed by Tim Buckwalter, this analyzer is downloadable from LDC[4] site. The text to be analyzed in BAMA should be transliterated into ASCII before any processing, and the results should be reconverted into Arabic to be intelligible. This well-known analyzer has been designed to be integrated into a machine translation application. It is highly cited in the literature and its source code is available. It contains a dictionary of lexicons of Arabic stems and lists of prefixes and suffixes. A list of rules that govern the compatibility of stems with affixes is also available.

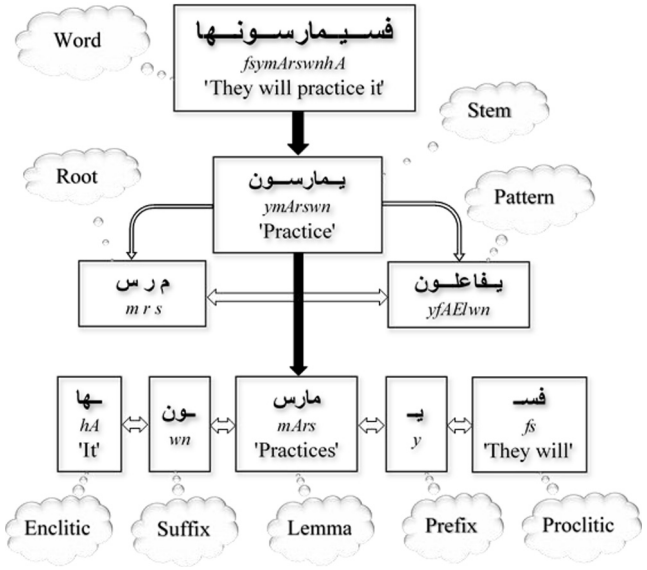

Figure 1. Segmentation of an Arabic written word.

- SAMA (Graff et al., 2010) is the latest version of BAMA. SAMA is an improved version of previous versions of BAMA. The set of words that this version is able to analyze is more consistent than that of older versions. In addition, the number of proposed solutions following the analysis of a word has increased significantly (Shah et al., 2010).
- MORPH2 (Kammoun et al., 2010) is based on a knowledge-based computational method. After identifying all the valid roots from all possible segmentations of the word on stem and affixes, an extraction step of morphosyntactic features based on research of possible vowelized forms of the word is performed.
- ALKHALIL1 (Boudlal et al., 2010) is an open source analyzer developed by the Arabic NLP team at the Mohammed first University (Morocco), in collaboration with ALECSO[5] and KACST.[6] For a given word, the analyzer provides all possible vowelized forms of the word. Each vowelized form is accompanied by several pieces of morphological information such as clitics, stem, root and POS Tag.
- MADAMIRA (Pasha et al., 2014) is a morphological analyzer that operates in the word context to assign the morphological tags for each word of the sentence. It is the result of combining two morphological analysis systems MADA (Habash et al., 2013, 2009) and AMIRA (Diab et al., 2007). The system analyzes at first the words of the sentence out of context using SAMA analyzer. To select one solution from among the multiple solutions obtained in the first phase, a disambiguation step based on the use of SVM and the language models is performed.

4. Specifications of Alkhalil2

This section is devoted to giving a global idea about the Alkhalil2 analyzer that we have developed. We first give a description of lexical resources. Then we explain the adopted method to segment words into clitics and stems. Finally, we present the list of different morphosyntactic tags that the system provides for each vowelized form of the word. An overview of the techniques used at each of these steps is given.

[4] https://www.ldc.upenn.edu/.

[5] http://www.alecso.org/site/.

[6] http://www.kacst.edu.sa/.

4.1. Used technical tools

As for the first version, the Alkhalil2 analyzer was developed with the object oriented language Java. Several reasons explain this choice. First, this language is highly portable. In addition, Unicode used by Java allows handling of the Arabic characters. Finally, a large community uses this language.

4.2. Linguistic resources

The construction and organization of the linguistic database are among the main tasks for the design of a morphological analyzer. We had begun by correcting errors in the database of the old version. Then, by testing the old version on a large corpus, we identified the missing data that had a negative impact on the performance of the analyzer. Therefore, we have integrated them in the database. Finally, we reorganized this database in order to optimize search in it and consequently make the system faster. We will present the main database files in this new XML version.

4.2.1. Exceptional file

This file contains 12 exceptional words (the word "الله" and its various forms obtained by concatenation with proclitics).

4.2.2. Clitic folder

It consists of two files:

- Proclitic file: it contains on the one hand the exhaustive list of simple proclitics, and on the other hand the list of compound proclitics resulting from the combination of simple proclitics. The 67 proclitics of this file are subsequently decomposed into three subclasses:
  - subclass labeled 'C' of proclitics that are compatible with all words,
  - subclass labeled 'N' of proclitics that can be concatenated only to nouns,
  - subclass labeled 'V' of proclitics that can be concatenated only to verbs.
- Enclitics file: it contains a list of 68 simple and compound enclitics. These enclitics are also classified, as in the case of proclitics, into three subclasses via the compatibility criterions.

4.2.3. Folder of non-derived words

This folder includes the non-derived words. It is constituted of two files:

- Proper noun file which is composed of 20,603 proper nouns.
- Functional word file, which contains 418 functional words such as prepositions, demonstrative pronouns and relative pronouns.

4.2.4. Folder of derived words

This folder is devoted to derived words. It is composed of two subfolders. The first one is reserved for verbs and the second one for derived nouns. The verb folder contains five files:

- The *VoweledStemCanonicPatternVerb* file which brings together 1,756 vowelized patterns relating to the stems of verbs.
- The *UnvoweledStemPatternVerb* file obtained by eliminating the diacritic marks in the previous file and keeping only the unrepeated patterns. This file contains 494 non vowelized patterns.
- The *VoweledLemmaCanonicPatternVerb* file that includes a set of 36 diacritized schemes related to the lemmas of verbs.
- The *RootVerb* file which contains 7502 roots. Each root is accompanied with their corresponding vowelized patterns, and each couple (root, vowelized pattern) is assigned the following morphological tags: part of speech and mood (indicative, subjunctive and jussive).

Similarly, the noun folder contains also five files:

- The *VoweledStemCanonicPatternNoun* file which is composed of 8042 vowelized patterns relating to the stems of nouns.
- The *UnvoweledStemCanonicPatternNouns* file which contains 1617 non vowelized patterns relating to the stems of nouns. These patterns are obtained by eliminating the diacritic marks in the previous file and keeping only the unrepeated patterns.
- The *VoweledLemmaCanonicPatternNoun* file that includes a set of 629 vowelized schemes related to the lemmas of nouns.
- The *RootNoun* file which contains 7692 roots. The roots are accompanied with their corresponding vowelized patterns. In addition, each couple (root, vowelized pattern) was assigned the following morphological tags: part of speech and case (nominative case, accusative case and genitive case).

Some of these patterns were obtained from the database of the open source Arabic morphology system Sarf.[7] The others were completed by members of our team.

To facilitate search in these files, we adopted a classification that takes into account both the word length (for roots and patterns) and the alphabetical order of the first letter of the word.

It remains to note that these bases can generate a rich set of 4,101,503 vowelized stems (2,197,962 stems relating to nouns and 1,903,541 stems relating to verbs).

4.3. Steps of analysis

Morphosyntactic analysis is carried out in the following five steps:

4.3.1. Preprocessing

To facilitate the subsequent steps, our method starts by preparing the input text. The system starts by segmenting the text into words. Thereafter, it normalizes these words by removing both kashida and diacritic marks. Moreover, any string of characters that is other than Arabic is also eliminated. Our analytical method stores in memory a complete copy of diacritic marks of input words (if they exist), in order to reject the results of analysis incompatible with these diacritic marks.

4.3.2. Segmentation

This step deals with the orthographic word obtained after preprocessing. The system regards it as a series of constituents (proclitic + stem + enclitic) and aims at identifying them. Thus, the system proposes all conceivable segmentations by browsing the proclitic and enclitic lists defined in Section 4.2.2. The system keeps only the segmentations that the associated proclitics and enclitics are compatible with.

4.3.3. Analysis of the stem

The diacritical marks being absent, the same stem can lead to various interpretations. First, it can be interpreted as a non-derived word. A second interpretation may refer to a derived noun and a third one to a verb. Consequently, for each segmentation validated in the previous step, the system performs a four-step analysis of the stem.

4.3.3.1. The stem as an exceptional word. The system checks whether the stem belongs to the list of exceptional words defined

[7] http://sourceforge.net/projects/sarf/.

in Section 4.2.1. In which case, the system assigns the exceptional word to the stem and stops the analysis. Otherwise, the system performs the remaining steps.

4.3.3.2. The stem as a non-derived word. The word is analyzed as being a non-derived word by checking whether the stem belongs to the non-derived class defined in Section 4.2.3. The segmentation is then accepted if the criteria of compatibility between the nature of the stem and that of clitics are valid. For valid segmentation, the system will provide the corresponding morphological features. Afterward, the system moves on to the next step.

4.3.3.3. The stem as a derived noun. The system checks whether the stem can be a derived noun. It first checks if the proclitic and the enclitic obtained during segmentation are noun-compatible, i.e. if they belong to class 'N' or to class 'C' (see Section 4.2.2). In such a case, the system identifies from the stem the possible roots and patterns following the steps below:

- using *UnvoweledStemPatternNoun* file defined in Section 4.2.4, we assign to the stem the reference patterns having the stem length;
- extracting the possible roots by identifying additional letters in the chosen patterns;
- making sure that the suggested root belongs to *RootNoun* file defined in 4.2.4;
- using the *RootNoun* file, check afterward that the root obtained from a pattern accepts the latter as the pattern of a possible derived form;
- assigning, in addition to the valid couple (root, pattern), the associated morphological tags and the possible diacritic marks to the studied stem. Such assignment is possible by using the *RootNoun* file.

4.3.3.4. The stem as a verb. Finally, the system checks if the stem is a verb stem. Such processing is similar to the previous one, except that verb files are used here.

Note that, to accelerate the process of analysis, the three later steps are done in parallel using the multi-threading.

4.3.4. Validation of results

The results obtained from the previous analysis will undergo the following validation processes:

1. Concordance between clitics and the output syntactic features:
   - to check the concordance of the ultimate character's diacritical mark of the stem with the proclitic syntactic function,
     - e.g.: the prepositions "ب"<*b*>and "ك"<*k*>appear only with nouns in genitive case.
   - to check the concordance of the part of speech with the enclitic,
     - e.g.: no concordance between the enclitic pronoun "هم"<*hm*>and passive verbs.
2. Concordance between the hamza allograph (ئ, إ, أ, ء or ؤ) in the proposed solutions and that of the input word,
   - e.g.: the short vowel dumma "ُ" cannot be followed by the hamza "ئ".
3. Concordance between the diacritic marks of the proposed solutions and those that may exist in the input word.

4.3.5. Display of the morphosyntactic analyzer's results

For a given word, Alkhalil2 analyzer enables thus the identification of the entire set of the possible solutions associated with their morphosyntactic features.

1. For nouns, these features are as follows:

(a) For non-derived nouns, the system gives:
- the vowelized form of the word
- the proclitic and the enclitic associated whenever they exist,
- the POS tags:
  - proper noun
  - functional word

(b) For derived words, the system generally proposes several solutions. For each of these solutions, the system outputs:
- the vowelized form of the word,
- the proclitic and the enclitic associated whenever they exist,
- the vowelized form of the stem and its pattern,
- the POS tags:
  - different verbal noun types,
  - active participle,
  - passive participle,
  - time and place nouns,
  - instrumental noun
  - gender (masculine or feminine)
  - number (singular, dual or plural)
- the root,
- the vowelized form lemma and its pattern,
- the case of the noun

2. For verbs, the system determines:
- the vowelized form of the word,
- the associated proclitic and enclitic whenever they exist,
- the vowelized form of the stem and its pattern,
- the POS Tags
  - tense of conjugation: imperfect, perfect, imperative,
  - active verb or passive verb
  - triliteral or quadrilateral verb,
  - augmented and unaugmented verb,
  - transitive or intransitive verb,
  - person conjugation.
- the root,
- the vowelized form of the lemma and its pattern,
- the mood of the verb.

3. For particles, the system determines the following features:
- vowelized forms of the particle;
- nature of the particle (particle of coordination, preposition etc...)

The analysis results are available in CSV, HTML and XML format.

5. Evaluation

To evaluate the performance of our analyzer Alkhalil2, we compare it to three other analyzers widely used in various applications of ALP. The first of them is the first version of the analyzer Alkhalil Morpho Sys. This comparison will allow us to measure the contribution of the database enrichment and the modifications carried out on the source code of the first version Alkhalil1. The second analyzer is the open source analyzer BAMA. The last is SAMA analyzer which is an improved version of BAMA analyzer.

To carry out this comparison, we used a large corpus of more than 72 million diacritized words. The latter consists of Tashkeela[8] corpus (63 million of diacritized words), Nemlar corpus (0.5 million of diacritized words) (Attiya et al., 2005) and a part of RDI[9] corpus not redundant with Tashkeela corpus (8.5 million of diacritized words). The Tashkeela and RDI corpora consist of diacritized texts from old classic books and some modern documents on subjects such as theology, grammar, history, economics and geography. The Nemlar corpus was produced and annotated by RDI, Egypt for the Nemlar Consortium. It consists of modern standard Arabic texts and covers several topics such as policy and general information.

We thus analyzed the non vowelized form of this corpus using the four analyzers. We were interested in the three common outputs shared by the four analyzers: namely the vowelized form of the word, the stem and the lemma (the lemma is not provided by Alkhalil1). The evaluation was conducted using several accuracy metrics:

- *Coverage:* percentage of words analyzed by the analyzer.
- *Speed:* number of analyzed words per second.
- *AN_Lemma:* average number of proposed lemmas per word.
- *AN_Stem:* average number of proposed stems per word.
- *AN_Diac:* average number of proposed vowelized forms per word (without the diacritic mark of the last character).

We present in Table 1 the values of these indicators for each analyzer.

We note that the best results are obtained with Alkhalil2. Indeed, this analyzer was able to analyze 99.31% of the words against only 90.18% for SAMA analyzer and a lower rate for the other two analyzers. This testifies to the great improvement made on Alkhalil1 analyzer and that is largely due to corrections made on its database and its enrichment. In addition, the high values of AN_Lemma, AN_Stem and AN_Diac obtained with Alkhalil2 reflect the richness of its database. Finally, Alkhalil2 analyzer achieves a speed close to that of the fastest analyzer (632 words per second against 685 for BAMA analyzer). However, the speed-coverage ratio is largely in favor of Alkhalil2 analyzer. This is a consequence of the richness of its database and their new organization that has allowed an optimal search.

The most used metrics to evaluate the accuracy of such analyzers are the precision, the recall and the F-measure. Calculating these metrics requires the availability of a corpus in which each word is accompanied with the set of all its possible features (e.g. for the lemma tag, every word must be accompanied by all its possible lemmas out of context). Such corpus does not exist as open source, it is not possible for us to calculate these indicators. However, each word in Nemlar corpus is accompanied by its three features determined in the word context: the lemma, the stem and the diacritized form. Therefore, we define the following metrics:

- *Rate_Lemma:* the rate of words whose associated lemma in the Nemlar corpus belongs to the set of suggested lemmas given by the analyzer.
- *Rate_Stem:* the rate of words whose associated stem in the Nemlar corpus belongs to the set of stems proposed by the analyzer.
- *Rate_Diac:* the rate of words whose associated vowelized form in the Nemlar corpus belongs to the set of vowelized forms given by the analyzer.

[8] http://sourceforge.net/projects/tashkeela/.
[9] http://www.rdi-eg.com/RDI/TrainingData/.

Table 1
Accuracy metric values for each analyzer.

	BAMA	SAMA	Alkhalil1	Alkhalil2
Coverage	80.13%	90.18%	88.51%	99.31%
Speed	685	336	23	632
AN_Lemma	2.5	2.47	Not given	4.71
AN_Stem	2.81	2.4	4.11	5.08
AN_Diac	2.91	6.51	8.07	8.05

Table 2
Indicator values for each analyzer.

	BAMA (%)	SAMA (%)	Alkhalil1 (%)	Alkhalil2 (%)
Rate_Lemma	78.34	91.14	Not given	97.16
Rate_Stem	79.65	91.36	81.31	96.76
Rate_Diac	79.98	91.50	86.79	97.21
Rate_Full	71.13	91.10	81.04	96.56

- *Rate_Full:* the rate of words whose three associated features in the Nemlar corpus (lemma, stem and vowelized form) belongs all to the set of features given by the analyzer.

Table 2 shows the values of these indicators for each analyzer applied on the non vowelized form of the Nemlar corpus.

The best results are obtained with Alkhalil2 analyzer. Indeed, the lemma of the word in the context provided by the Nemlar corpus is among the lemmas proposed by Alkhalil2 analyzer for 97.16% of the words against only 91.14% of the words for SAMA analyzer. The same remarks can be made for the other two features. We also note that the results obtained with BAMA and Alkhalil1 analyzers are low compared to those of the other two analyzers. Finally, the list of potential results provides by Alkhalil2 analyzer contains in 96.56% of words the three features assigned to words in the Nemlar corpus, while this proportion decreases to 91.10% with SAMA analyzer. This demonstrates the robustness and accuracy of our analyzer.

6. Conclusion

In this article, we illustrated the various stages of the new version of Alkhalil analyzer. We presented its database and focused on corrections and improvements we have made on this database. The comparison conducted on a representative corpus has showed that improvements on the old version of Alkhalil have significantly ameliorated the performance of this new version. Furthermore, the comparison with two morphological analyzers among the most cited in the literature has demonstrated the superiority of our analyzer. The analyzer provides also the following functionalities:

- Ability to search by the root: when the user enters a root, the program displays all the words in the text with this root as possible root, in addition to the location of the word in the text and its context.
- Indexing: the program indexes every word in the text by specifying its occurrence frequency and their locations in the text.

This analyzer was used in several morphological disambiguation systems. Indeed, (Chennoufi and Mazroui, 2016) used the Alkhalil2 analyzer to develop an Arabic vowelization system. Similarly, (Ababou and Mazroui, 2016) also developed an Arabic POS Tagger by using Alkhalil2 analyzer during the morphological phase.

References

Ababou, N., Mazroui, A., 2016. A hybrid Arabic POS tagging for simple and compound morphosyntactic tags. Int. J. Speech Technol. 19, 289–302. http://dx.doi.org/10.1007/s10772-015-9302-8.

Al-Sughaiyer, I.A., Al-Kharashi, I.A., 2004. Arabic morphological analysis techniques: a comprehensive survey. J. Am. Soc. Inf. Sci. Technol. 55, 189–213. http://dx.doi.org/10.1002/asi.10368.

Attiya, M., Yaseen, M., Choukri, K., 2005. Specifications of the Arabic written corpus produced within the NEMLAR project.

Boudlal, A., Lakhouaja, A., Mazroui, A., Meziane, A., Ould Abdallahi Ould Bebah, M., Shoul, M., 2010. Alkhalil Morpho SYS1: a morphosyntactic analysis system for arabic texts. In: International Arab Conference on Information Technology. Benghazi, Libya, pp. 1–6.

Buckwalter, T., 2002. Arabic Morphological Analyzer Version 1.0. Linguist. Data Consort. n° LDC2002L49.

Chennoufi, A., Mazroui, A., 2016. Impact of morphological analysis and a large training corpus on the performances of Arabic diacritization. Int. J. Speech Technol. 19, 269–280. http://dx.doi.org/10.1007/s10772-015-9313-5.

Cohen, D., 1970. Essai d'une analyse automatique de l'arabe. Etudes de Linguistique Sémitique et Arabe. The Hague, Paris, pp. 49–78.

Darwish, K., Oard, D.W., 2002. Term selection for searching printed Arabic. Proc. 24th Annu. Int. ACM-SIGIR Conf. (SIGIR 2002), pp. 261–268. http://dx.doi.org/10.1145/564422.564423.

Diab, M., Kadri, H., Daniel, J., 2007. Automated methods for processing Arabic text: from tokenization to base phrase chunking. Arabic Computational Morphology: Knowledge-Based and Empirical Methods.

Farghaly, A., Shaalan, K., 2009. Arabic natural language processing: challenges and solutions. ACM Trans. Asian Lang. Inf. Process. 8, 1–22. http://dx.doi.org/10.1145/1644879.1644881.

Graff, D., Maamouri, M., Bouziri, B., Krouna, S., Kulick, S., Buckwalter, T., 2010. Standard Arabic Morphological Analyzer (SAMA).

Habash, N., Rambow, O., Roth, R., 2009. MADA + TOKAN: a toolkit for Arabic tokenization, diacritization, morphological disambiguation, POS tagging, stemming and lemmatization, Proc. Second Int. Conf. Arab. Lang. Resour. Tools, pp. 102–109.

Habash, N., Roth, R., Rambow, O., Eskander, R., Tomeh, N., 2013. Morphological analysis and disambiguation for dialectal Arabic. Hlt-Naacl, 426–432.

Kammoun, N., Belguith, L., Hamadou, A., 2010. The MORPH2 new version: a robust morphological analyzer for Arabic texts. 10th International Conference Journées d'Analyse Statistique Des Données Textuelles. Sapienza University of Rome.

Pasha, A., Al-badrashiny, M., Diab, M., Kholy, A., El Eskander, R., Habash, N., Pooleery, M., Rambow, O., Roth, R.M., 2014. MADAMIRA : a fast, comprehensive tool for morphological analysis and disambiguation of Arabic. Proc. 9th Lang. Resour. Eval. Conf., pp 1094–1101.

Sawalha, M., Atwell, E., Abushariah, M.A.M., 2013. SALMA: Standard Arabic Language Morphological Analysis. Proc. ICCSPA Int. Conf. Commun. Signal Process. their Appl. Sharjah, UAE 1-6. http://dx.doi.org/10.1109/ICCSPA.2013.6487311.

Shah, R., Dhillon, P.S., Liberman, M., Foster, D., Maamouri, M., Ungar, L., 2010. A new approach to lexical disambiguation of Arabic text. 10 Proc. 2010 Conf. Empir. Methods Nat. Lang. Process, pp. 725–735.

Soudi, A., Neumann, G., Bosch, van den, A., 2007. Arabic computational morphology: knowledge-based and empirical methods. Arabic Computational Morphology. Springer, Netherlands, Dordrecht, pp. 3–14. http://dx.doi.org/10.1007/978-1-4020-6046-5_1.