Journal of Information Systems Engineering and Management
2025, 10(4)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Comprehensive Analysis of Arabic Tokenization System Preprocessing using the Matching Model

Dr. Ibrahim Abdelfattah Almajali[1], Dr. Mutlaq Moraya Nafah Alharbi[2]
[1]Art college, Department of Arabic language, King Faisal Uuniversity, Alhafof ,The Eastern Province, Saudi Arabia
ialmajali@kfu.edu.sa
[2]Ministry of Education, matlaqalharbi49@gmail.com

ARTICLE INFO

Received: 10 Nov 2024

Revised: 25 Dec 2024

Accepted: 26 Jan 2025

ABSTRACT

This research paper proposes a novel Arabic word tokenization system based on the knowledge Word tokenization is the first stage for higher-order Natural Language Processing (NLP) tasks like Part-of-Speech (PoS) tagging, parsing, and named entity recognition. The amount of text on the World Wide Web is growing daily in the present era of technology, necessitating the use of advanced instruments. Since more and more people speak Arabic around the world, Arabic language processing systems must be improved. Due to the writing style of Arabic with a lack of support for capitalization features and the use of compound words, it is difficult to perform word tokenization. Arabic's inconsistent usage of space between words makes it difficult to tokenize words because of its cursive form. Word tokenization plays a vital role in all aspects of natural language processing. Different applications can be created once words have been tokenized. To develop this system, a maximum matching model with its two variations, namely forward and reverse maximum matching is used. The proposed system is implemented in Python. The results produced during system evaluation report high performance.

Keywords: Natural Language Processing, Arabic word tokenization, Arabic Language Processing, PoS Tagging, Maximum Matching Model.

1 Introduction

Word tokenization is a vital task of Natural Language Processing (NLP) which is the sub-field of Artificial Intelligence that enables computer systems to interact and behave like human beings (1). Practically every language spoken on the planet might benefit from research in the field of NLP. In NLP, computers are set up to successfully capture and manipulate human language. NLP experts are seeking to convey information about how humans acquire and use common language. They employ advanced tools and techniques that can be creatively advanced to build computer frameworks that learn and operate natural language and do the required tasks (2). The basis for NLP can be found in a variety of fields, including artificial intelligence (AI), data sciences, electronic and electrical design (3, 4). Applications for NLP include a wide range of topics, including word tokenization, discourse recognition, client interface, Cross-Language data Information Recovery (CLIR), and content preparation and summary. Word tokenization plays a vital role in all aspects of natural language processing. Different applications can be created once words have been tokenized (5). Word tokenization is challenging for computers but significantly simpler for native speakers (6).

Arabic's inconsistent usage of space between words makes it difficult to tokenize words because of its cursive form. Written text can be separated and isolated into discrete pieces called words thanks to word tokenization. Word tokenization is a technique for defining the boundaries of words in spoken languages. Part of speech (POS) is a good example of NLP, as are marking morphological analysis, identifying designated entities (NER), removing words, and the importance of shallow parsing in all NLP systems (7). The tokenization of Arabic words is particularly challenging since, unlike other languages, the space character is not only rarely used as a delimiter. In the Arabic language, the use of space makes space absence and addition problematic. The Arabic-English translation system is used to view this type of tokenization in Arabic text (8) due to the space deletion problem, for example, the Arabic word " دودة الأرض" would become "دودةالأرض" if the space between " دودة " and "الأرض " was deleted.

The rest of the article is divided into sections that discuss the literature review in section 2, the characteristics of the Arabic language in section 3, difficulties in Arabic word tokenization in section 4, a proposed architectural design in section 5, and the study's conclusion in the last section.

2 Literature Review

Tokenization is a necessary first step in the further processing of any natural language. Arabic tokenization has been discussed in numerous studies and applied in numerous systems. These solutions include morphological analysis (9, 10), discretization (11), Information Retrieval (12), and POS Tagging (13, 14).

Attia (15) described a rule-based tokenizer that handled tokenization as a complete-rounded process with a preprocessing stage (white space normalizer), and a post-processing stage (token filter). The author also handled multiword expressions and ambiguous words. Farasa (16) is a brand-new Arabic segmenter that employs SVM for ranking. On common MT and IR tasks, they compared the proposed text segmenter to segmenters from Stanford and MADAMIRA. They found that Farasa performed much better (in terms of accuracy) than both on the IR tasks and was on par with MADAMIRA on the MT tasks.

Habash and Rambow (14) suggested a morphological analyzer for tokenizing and morphologically tagging Arabic words. They learned how to use these classifiers to choose items from the output of the analyzer, as well as how to classify each unique morphological characteristic individually. On all tasks, they attained accuracy rates in the upper nineties.

Benajiba and Zitouni (17) proposed two segmentation systems, morphological segmentation, and Arabic Treebank segmentation, and illustrated their effects on mention identification, a crucial job in natural language processing. Research on the Arabic TreeBank corpus demonstrates morphological segmentation with high accuracy.

Aliwy (18) suggested a hybrid unsupervised method for the Arabic tokenization system. They first segmented phrases into words, and then they used the author's analyzer to generate every possible tokenization for each word. Then, to clear up the ambiguity, written rules, and statistical techniques are used. Each word receives a tokenization as the output. The statistical approach was used in which text is manually tokenized from the Al-Watan 2004 corpus and trained on a variety of terms and reports high accuracy.

3 Challenges in Arabic Word Tokenization

Arabic is one of the Semitic languages, along with Hebrew, Aramaic, and Amharic. It serves as a common language for a sizable population. About 400 million people speak Arabic as their first language, according to estimates (12, 19). Since it is used for religious education in Islam, many other speakers from different countries at least have a passing familiarity with it. Arabic is the fifth most frequently spoken language in the world and one of the six official languages of the United Nations (20, 21).

In Arabic, words are separated by white spaces and other punctuation, while sentences are delimited by periods, dashes, and commas. While Arabic numbers are written and read from left to right, Arabic script is written from right to left. Arabic has two different sorts of symbols (12, 22). Nouns, verbs, and particles are the three primary components of speech used to categorize Arabic words. Arabic words can be either feminine or masculine. In literary Arabic, the same feature is used to denote numbers (singular, dual, and plural) in both nouns and verbs.

Arabic morphology is very complex. Words are created by deriving their roots into patterns. There are three grammatical cases in Arabic as well. Nominative, accusative, and genitive are these cases. The uncertainty introduced by morphology renders the exact keyword-matching process useless for retrieval. Morphological ambiguity can show up in a variety of situations. Clitics, for instance, may unintentionally create a form that is homographic or homogeneous with another full word (the same word with two or more alternative meanings) (12, 15, 20).

Besides the complex morphology, Arabic has also a complex type of plurals which is known as broken plural. Plurals in Arabic do not obey morphological rules. Unlike English, the plural in Arabic indicates any number higher than two. Arabic has also very diverse types of orthographic variations. They are very common and present real challenges for Arabic NLP systems.

Over the last decades, Arabic has become one of the popular areas of research in information retrieval, especially with the explosive growth of the language on the Web, which shows the need to develop good techniques for the increasing contents of the language. This increasing interest in Arabic, however, is caused by its complex morphology, which is radically different from the European and East Asian languages (20). Additionally, Arabic is exceedingly rich in its derivational system and contains complex grammatical rules (12). These characteristics make the language difficult to process computationally and to analyze morphologically since, in the majority of circumstances, accurate keyword matching between documents and user queries is insufficient.

4 Maximum Matching Model

The maximum matching technique is used for several NLP applications, especially for word tokenization (23, 24). It is a rule-based technique for longest matching to accomplish the desired results (25). Through the application of the maximum matching technique, word sense information is used. The longest matching approach is used by the maximum matching algorithm. The data string is matched with a dictionary passage using the maximum matching technique, and the optimum tokenization arrangement is chosen using the shortest and longest words (26, 27). This algorithm searches the longest matching term in a left-to-right direction (suitable for Arabic material). If the sentence contains words with only one character, this algorithm will only provide one type of arrangement. The coming about phrase tokenization is continually sub-optimal because the calculation locally determines the sections.

Our proposed approach is based on an Arabic word tokenization model that uses word sense data. To complete the work at hand, the maximum matching technique is used. The dataset of Arabic word sense information will be used by this method. The longest and shortest word from among all possible tokenized arrangements is selected in the Maximum Matching algorithm, which matches character strings with dictionary passages (28). The program looks for the longest matching word and operates from right to left (suitable for Arabic text). If the sentence contains words with a single character, this algorithm will provide a unique arrangement. Due to the algorithm's local section determination, there is less need for sentence tokenization to occur. FMM (Forward Maximum Matching) and RMM (Reverse Maximum Matching) are two methods via which this algorithm operates. While the characters in RMM are checked from left to right, those in FMM are checked from right to left. In our research, we mostly use the maximum matching model because it uses fewer resources than machine learning approaches, a features file is not required, and pre-labeled datasets are not required. The proposed mechanism for Arabic word tokenization is shown in Figure 1.

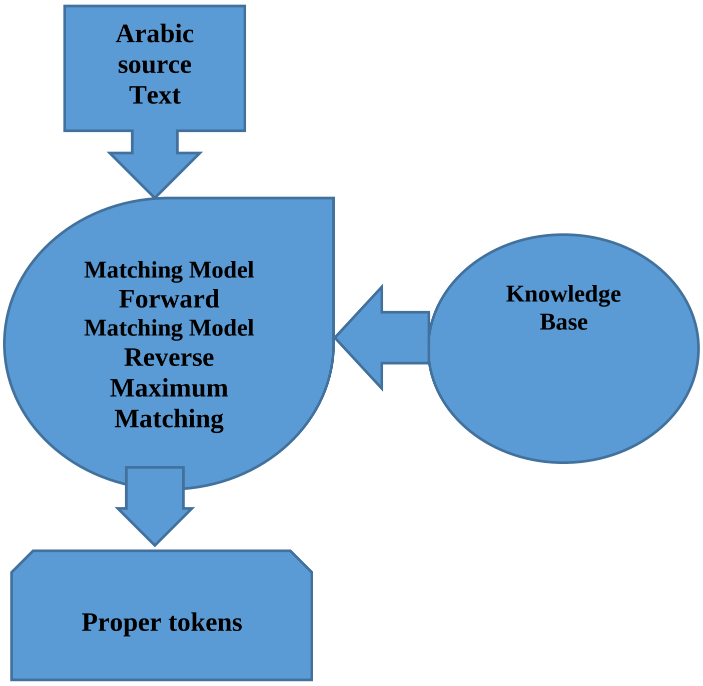
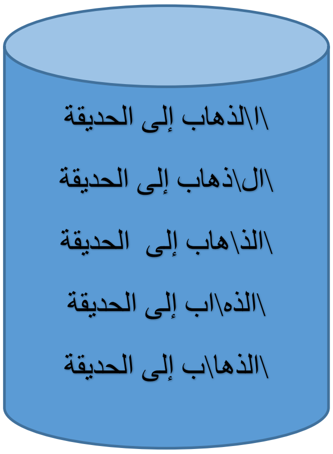

Arabic source Text
Matching Model Forward Matching Model Reverse Maximum Matching
Knowledge Base
Proper tokens

Figure 1. alleged Technique

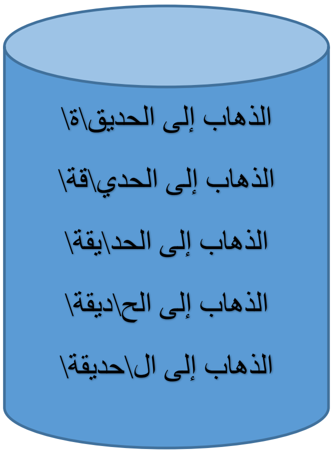
The alleged matching model techniques combined the results of both Forward Maximum Matching (FMM) and Reverse Maximum Matching (RMM) techniques. An Arabic dictionary is used for forward and reverse maximum matching (29). These techniques receive a single sentence as input and output a tokenization of each word (30). Before being tokenized using RMM, the words are first tokenized using FMM. The output is then produced by matching the FMM and RMM results. If they generate the same results then the tokenization is clear. If the FMM and RMM results do not agree, the phrase will be declared unclear.

Tokenization is carried out from right to left when using FMM by considering the longest maximum matching and verifying each character separately (24, 31, 32). For instance, the following text tokenizes the Arabic words using the FMM approach.

الذهاب إلى الحديقة meaning “going to park”



In RMM, word tokenization is started from left to right direction (33). During tokenization, the last character of the word is selected and then matched against the stored knowledge. Considering the previous example, the RMM approach is shown below:



5 Evalnation and Results

A simple Arabic dataset is used to test the proposed mechanism. Precision, recall, and F-measure (F-score) are calculated to evaluate the Arabic tokenization system. Precision is the similarity of two or more evaluations. Precision and recall have the opposite relationship; as accuracy increases, memory declines. The harmonic mean of Precision and Recall is used to calculate the F-measure. The dataset is collected from various Arabic newspapers and online websites.

Accuracy represents the number of correctly tokenized text over the total number of text instances.

$$\text{Accuracy} = (\text{TN} + \text{TP})/(\text{TN} + \text{FP} + \text{TP} + \text{FN}) \qquad (1)$$

Where TN = True Negative, FP = False Positive, TP = True Positive and FN = False Negative

As the accuracy is not a good metric when the dataset is unbalanced the accuracy of the system is not calculated here for Arabic text tokenization.

Precision is a measure of how many of the positive predictions about the tokens made are correct (true positives).

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \qquad (2)$$

Recall is a measure of how many of the positive cases the system correctly predicted, over all the positive cases in the text. It is sometimes also referred to as Sensitivity.

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}) \quad (3)$$

F Score is the combination of precision and recall. It is the harmonic mean of the two values. The harmonic mean is just another way to calculate an average which is generally described as more suitable for ratios than the traditional arithmetic mean.

$$\text{F Score} = 2 \times (\text{Precision} \times \text{Recall})/(\text{Precision} + \text{Recall}) \ (4)$$

The proposed system is created using PyCharm which is an integrated development environment (IDE) for Python programming language. Precision, Recall, and F-measure (F-score) evaluations have been used to determine the overall effectiveness of the proposed system. The values for Accuracy, Recall, and F-score for the tested Arabic text are displayed below Table 1.

Table 1: Recall, Precision, and F-score measures for Arabic text tokenization

Source	Words	Precision	Recall	F-score
Al-Bayan[1]	1600	96%	52%	987%
Al Anbaa[2]	1150	95%	43%	98%
Al Watan[3]	1110	96%	41%	97%
Asharq Al-Awsat[4]	1220	96%	52%	97%

6 Conclusion

In this work, a ruled-based method is discussed for addressing Arabic word tokenization. Due to the spacing issues between the words of Arabic language, it become difficult to tokenize its text. A novel technique for tokenizing Arabic text is introduce which keep track of space insertion and deletion, compound words, and reduplicated words to increase the performance of tokenizer. esults indicate that the proposed mechanism produce better results while calculating precision, recall and F-score. In the future, deep learning techniques will be applied for the Arabic text tokenization including deep convolution neural networks and recurrent neural networks.

conflict of interest

The author declare no conflict of interest

Funding: This work was supported by the Deanship of Scientific Research, King Faisal University, Saudi Arabia Grant No. KFU251752

[1] https://www.albayan.ae/
[2] https://www.alanba.com.kw/newspaper/
[3] https://www.alwatan.com.sa/
[4] https://aawsat.com/

References

[1] Grefenstette G. Tokenization. Syntactic Wordclass Tagging. 1999:117-33.

[2] Derczynski L, Kjeldsen AS, editors. Bornholmsk natural language processing: Resources and tools. Proceedings of the Nordic Conference of Computational Linguistics (2019); 2019: Linköping University Electronic Press.

[3] Rubin VL, Chen Y, Conroy NK. Deception detection for news: three types of fakes. Proceedings of the Association for Information Science and Technology. 2015;52(1):1-4.

[4] Chowdhary K, Chowdhary K. Natural language processing. Fundamentals of artificial intelligence. 2020:603-49.

[5] Rashid R, Latif S, editors. A dictionary based Urdu word segmentation using maximum matching algorithm for space omission problem. 2012 International Conference on Asian Language Processing; 2012: IEEE.

[6] Mustafa M, Eldeen AS, Bani-Ahmad S, Elfaki AO. A comparative survey on Arabic stemming: approaches and challenges. Intelligent Information Management. 2017;9(02):39.

[7] Abbas Q, editor Semi-semantic part of speech annotation and evaluation. Proceedings of LAW VIII-The 8th Linguistic Annotation Workshop; 2014.

[8] Alginahi YM. A survey on Arabic character segmentation. International Journal on Document Analysis and Recognition (IJDAR). 2013;16(2):105-26.

[9] Beesley KR, editor Finite-state morphological analysis and generation of Arabic at Xerox Research: Status and plans in 2001. ACL Workshop on Arabic Language Processing: Status and Perspective; 2001: Citeseer.

[10] Buckwalter T. Buckwalter Arabic morphological analyzer version 1.0. Linguistic Data Consortium, University of Pennsylvania. 2002.

[11] Nelken R, Shieber S, editors. Arabic diacritization using weighted finite-state transducers. Proceedings of the 2005 ACL Workshop on Computational Approaches to Semitic Languages; 2005: Association for Computational Linguistics.

[12] Darwish K, Magdy W. Arabic information retrieval. Foundations and Trends® in Information Retrieval. 2014;7(4):239-342.

[13] Diab M, Hacioglu K, Jurafsky D, editors. Automatic tagging of Arabic text: From raw text to base phrase chunks. Proceedings of HLT-NAACL 2004: Short papers; 2004.

[14] Habash N, Rambow O, editors. Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop. Proceedings of the 43rd annual meeting of the association for computational linguistics (ACL'05); 2005.

[15] Attia M, editor Arabic tokenization system. Proceedings of the 2007 workshop on computational approaches to semitic languages: Common issues and resources; 2007.

[16] Abdelali A, Darwish K, Durrani N, Mubarak H, editors. Farasa: A fast and furious segmenter for arabic. Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Demonstrations; 2016.

[17] Benajiba Y, Zitouni I, editors. Arabic Word Segmentation for Better Unit of Analysis. LREC; 2010: Citeseer.

[18] Aliwy AH. Tokenization as preprocessing for Arabic tagging system. International Journal of Information and Education Technology. 2012;2(4):348.

[19] Mirkin B. Population levels, trends and policies in the Arab region: challenges and opportunities: United Nations Development Programme, Regional Bureau for Arab States USA; 2010.

[20] Mustafa M, Suleman H, editors. Mixed language Arabic-English information retrieval. Computational Linguistics and Intelligent Text Processing: 16th International Conference, CICLing 2015, Cairo, Egypt, April 14-20, 2015, Proceedings, Part II 16; 2015: Springer.

[21] Chung W. Web searching in a multilingual world. Communications of the ACM. 2008;51(5):32-40.

[22] Habash N, Rambow O, editors. Arabic diacritization through full morphological tagging. Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers; 2007.

[23] Gai RL, Gao F, Duan LM, Sun XH, Li HZ, editors. Bidirectional maximal matching word segmentation algorithm with rules. Advanced materials research; 2014: Trans Tech Publ.

[24] Zhao Y, Li H, Yin S, Sun Y. A New Chinese Word Segmentation Method Based on Maximum Matching. J Inf Hiding Multim Signal Process. 2018;9(6):1528-35.

[25] Chen M-h, Yin J-h, Shu C, Wang M-j. A chinese word segmentation system design based on forward-backward maximum matching algorithm. Information Technology. 2009(6):124-7.

[26] Cheung A, Bennamoun M, Bergmann NW. An Arabic optical character recognition system using recognition-based segmentation. Pattern recognition. 2001;34(2):215-33.

[27] Elnagar A, Al-Debsi R, Einea O. Arabic text classification using deep learning models. Information Processing & Management. 2020;57(1):102121.

[28] Alharbi A, Kalkatawi M, Taileb M. Arabic sentiment analysis using deep learning and ensemble methods. Arabian Journal for Science and Engineering. 2021;46:8913-23.

[29] Shaalan K. A survey of arabic named entity recognition and classification. Computational Linguistics. 2014;40(2):469-510.

[30] Ahn KJ, Guha S. Linear programming in the semi-streaming model with application to the maximum matching problem. Information and Computation. 2013;222:59-79.

[31] Wang R, Luan J, Pan X, Lu X. An improved forward maximum matching algorithm for Chinese word segmentation. Jisuanji Yingyong yu Ruanjian. 2011;28(3):195-7.

[32] Wong P-k, Chan C, editors. Chinese word segmentation based on maximum matching and word binding force. COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics; 1996.

[33] Zhang L, Li Y, Meng J, editors. Design of Chinese word segmentation system based on improved Chinese converse dictionary and reverse maximum matching algorithm. Web Information Systems–WISE 2006 Workshops: WISE 2006 International Workshops, Wuhan, China, October 23-26, 2006 Proceedings 7; 2006: Springer.