Integrated approaches in a morphological analyzer of the Arabic language

Said Iazzi[1], Abderrazak Iazzi[1], Hicham Gueddah[2], Abdellah Yousfi[3], Mostafa Bellafkih[4]
[1]Laboratory of Research in Computer Science and Telecommunications, Informatics Department, Faculty of Sciences, Mohammed V University in Rabat, Rabat, Morocco
[2]Intelligent Processing and Security of Systems Team, Informatics Department, FSR, Mohammed V University in Rabat, Rabat, Morocco
[3]FSJES-Souissi, Mohammed V University in Rabat, Rabat, Morocco
[4]STRS Laboratory, Department Telecommunications Systems, Networks and Services, National Institute of Posts and Telecommunications, Rabat, Morocco

Article Info

Article history:

Received Jul 21, 2022
Revised Nov 10, 2022
Accepted Nov 19, 2022

Keywords:

Databaseofsurface pattern
Finite state automaton
Morphological analysis
Morphological corpus
Table correspondence
Viterbi algorithm

ABSTRACT

This article presents a systemof a morphological analyzer of the Arabic language, by integrating several approaches and the viterbi algorithm. First approach is based on database for all thesurface patterns in the Arabic language, second approach is Buckwalter Arabic morphological analyzer and the last approach is based on finite state automaton. With the integration of correspondence tables between affixes in these approaches. The combination between these approaches in our analyzer is very important. Our analyzer is tested on a morphological corpus of 200,000 words, which generalize the words of the Arabic language. The effectiveness of the proposed approaches is demonstrated experimentally and the results obtained are comparable to the state of the art. Moreover, it shows the interest and the advantages of integrating these approaches are to improve our morphological analyzer.

Corresponding Author:

Said Iazzi
Laboratory of Research in Computer Science and Telecommunications, Informatics Department
Faculty of Sciences, Mohammed V University
Rabat, Morocco
Email: iazzisaid@yahoo.fr

1. INTRODUCTION

Morphological analysis is a very important step in various applications of natural language processing [1], [2]. Integration of approaches for morphological analyzer of the Arabic language is necessary [3], it requires the development of algorithms that can interpret and analyze word structure at many levels [4], such as processing linguistic rules, patterns of Arabic words and data dictionary, etc. Morphological analysis is used in a variety of applications of natural language processing [5], amongthem: information retrieval and extraction, machine translation, text mining, machine synthesis and Arabic learning systems [6].

The morphological analysis of Arabic is very complicated in the automatic processing because of the structure of complex word where we have stems, infixes, prefixes, suffixes, and complex patterns [7], [8]. It detects the different morphological entities in the word and provides a morphological representation. More, for each prefix or suffix can have its own syntactic attachment; this means that we have the resources to use the results of the morphological analysis stage in the higher stages of Arabic processing as syntactic analysis and error processing.

In recent years, several works have been developed in the axis of morphological analysis of the arabic language which are generally based on one of the following approaches: first approach based on linguistic rules [9]-[11], the second approach based on dictionary-based [10], [11], the following approach based on a word pattern [12]-[15], the fourth approach based on finite state automaton [16]-[18] andthe last approach is hybrid approach which combines its different approaches [19].

In this work, we have proposed a study of approaches based on surface pattern and on finite state automaton. This allowed detecting the types of errors and the strengths and weaknesses of each analyzer. Subsequently, it will be very interesting to combine these approaches in a single analyzer to increase both the precision and the recall and to decrease the execution time compared to our first analyzers [20]. It is very important to combine several approaches to process and analyze words, in the arabic language, several analyzers have been developed, we can cite, for example, that of [10], [12], [18], [21]. In this article, we propose an integration of several approaches to build an Arabic morphological analyzer that meets all needs.

In sub-section 2.1, we will present a look on ourmorphological analyzer based on surface patterns. Sub-section 2.2, we will present an overview on our second morphological analyzer based on finite state automate and viterbi algorithm [22]. Section 3, we will present an integration of several approaches in order to build a morphological analyzer that deals with all cases of Arabic words. In section 5, we describe our morphological analyzer with tests and results. We provide our method for evaluating the approaches in the previous sections. In the last section, we conclude this work with some conclusions and recommendations.

2. METHOD

2.1. Morphological analysis based on thesurface patterns

We have developed an approach to improve our morphological analysis which is based on the surface pattern of arabic language words. It is mainly based on the construction of the surface patterns database. This morphological analyzerdetermines one or many possible patterns for a given word, in order to find all possible analyzes of this word.

Patterns allow effectively modeling morphological variations within words and detecting the root of a word. in this axis, several works have been developed which use the pattern-root approach, among which we cite [12], [14]. All these works use for the morphological analysis of words the classical patterns of Arabic words. In our morphology analyzer, we use a new adapted pattern that we called surface patterns.

To build the database of surface patterns of Arabic words: For exemple the classical pattern of the word (جادوا) is (فعلوا), but its surface pattern is (فالوا). The algorithm we used to build surface patterns from a word: For a word$w = l_1 l_2 \ldots l_n$ (l_n Character of the wordw) and R its root. The surface patternsofwordwis$p = f_1 f_2 \ldots\ldots f_n$ where:

$$\begin{cases} \text{f}_\text{i} \text{ is one of three letters "ف،ع،ل"if} \text{l}_\text{i} \in \text{R and } \text{l}_\text{i} \notin \{\text{ا،و،ي}\} \\ \text{f}_\text{i} = \text{l}_\text{i} \text{ if } \text{l}_\text{i} \text{ is not in R Where } \text{l}_\text{i} \in \{\text{ا،و،ي}\} \end{cases}$$

relying on the surface patternsdatabase approach, for the word "قائلون" the root is "قال" and the surface pattern is "فائلون". The surface patterns of the root$R = g_1 g_2 \ldots g_k$ (gi is a character root) is$P' = f'_1 f'_2 \ldots f'_k$ with:

$$\begin{cases} \text{f}'_\text{i} = \text{one of three lettres (ف، ع، ل) if } \text{g}_\text{i} \text{is a non variant letter in R} \\ \text{f}'_\text{i} = \text{g}_\text{i} \text{ if not} \end{cases}$$

non-variant letter in a root R is a letter that staysthe same when generating words from that root.

To perform out the morphological analysis with a word w by the surface patterns approach, we go through the following steps:

$$f(m; w) = \sum_{i=1}^{N} 1_{[m_{i;}\, w_i]}$$

we keep just the surface patterns having $f \neq 0$.

- Extractiononly of the surface patterns of the solution roots from the surface patterns of the word analyze.
- Construction of roots from surface patterns, roots associated with word w and andverification whether these roots exist in the root database or not.
- To test and evaluate our approach, we have constructed all surface patterns of words derived from the Arabic language. This step was handled by a group of Arabic language linguists. They used a set of Arabic language references.

Example:

The phases of analysis of the word "نقول":

- Searching for the surface patterns corresponding to the word"نقول". We find these surface patterns: P_1="نفول"; P_2= "فعول".
  a. "نفول"from the root"قال".
  b. "فعول" from the root"نقل".
- Extraction the surface patterns of the roots of P_1 and P_2,we findbothsurface patterns of the roots SR_1="فال" and SR_2="فعل".
- Construction of the roots SR_1, SR_2, from P_1, P_2, and word W. We find the following root solutions: R_1="قال" and R_2="نقل".

For our surface patterns based analyzer (Figure 1). We used the following sources:

- Lexicon of 6,216 surface patterns. This lexicon contains all the morphological classes of words derived from the arabic language.
- Root dictionary containing 1,200 roots.
- Radical dictionary containing 6,000 radicals.

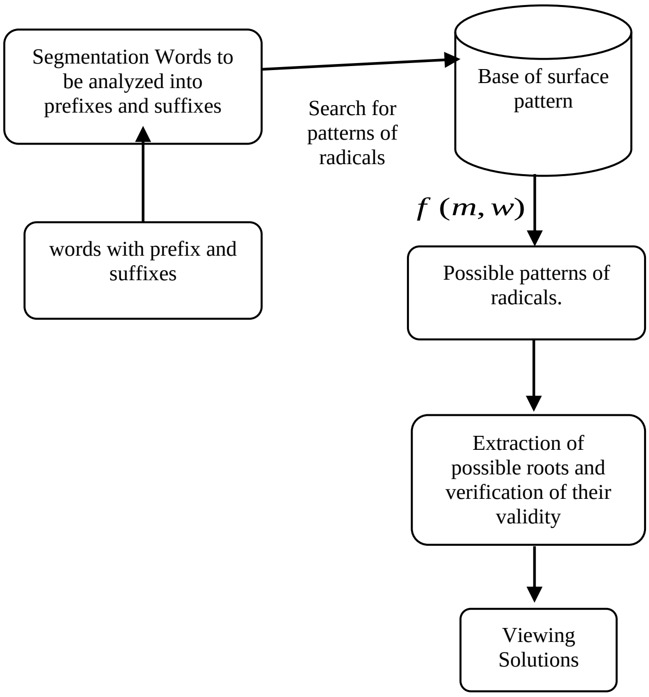
Segmentation Words to be analyzed into prefixes and suffixes
Base of surface pattern
Search for patterns of radicals
words with prefix and suffixes
f (m, w)
Possible patterns of radicals.
Extraction of possible roots and verification of their validity
Viewing Solutions

Figure 1. The steps of our morphological analyzer of Arabic words who use surface patterns

2.2. Morphological analyzer based on finite state automaton

The finite state automaton is an adaptive automaton which successively changes its structure according to the application of adaptive actions associated with the transition rules carried out by the automaton. The finite state automaton he has great potential for to be used in natural language processing [16]. It able to simplify and represents complex linguistic situations such as ambiguities and non-determinisms especially in the Arabic language. Additionally, the recognition formalism can be put in place for a recognition formalism can be implemented for pre-processing texts for a variety of scenarios such as morphologicalanalysis, syntactic verification, text interpretation, automatic translation and computer-assisted language learning [7], [23]. The form of the finite state automaton or adaptive automaton makes it possible to process the different classes of languages.

The finite state automaton analyzer [17] is an analyzer where each word of the Arabic language is represented by a path in this finite state automaton. To analyze a word, the finite-state automaton analyzer goes through the following two steps; The finite state automaton analyzer [24] is a morphological word analyzer, where every word of the Arabic language is represented by paths of the Arabic alphabets in this finite state automaton. For morphological analysis of a word, the finite-state automaton analyzer goes through the following two steps:

- Construction of a network of all the words of the Arabic language.
- Search the possible solutions for our analyzer in this global network.

This system based on very restricted dictionaries and searches the solutions in the global network using the Viterbi algorithm, and each word is modeled by a path, whose radical letters are presented by a state which loops on itself, ame the affixes are presented by the characters forming the affixes.
Example: For the words 'فجامعها' , 'فداخلها'…, are presented by the following diagram (Figure 2).

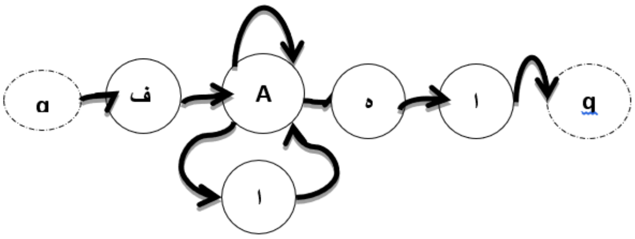

Figure 2. Finite state automaton of words 'فداخلها' ,'فجامعها'

Based on all the affixes of the Arabic language, we build the global network.

Our network is defined entirely by (Figure 3):

- The set of all the states is Q,it consists of all the characters composing the affixes (suffixe, prefixe and infixe), of state A, the start state q_I and the final state q_F:

  Q={q_I, q_F, A,"ف","و","ي","ل",…,"ه","م","ت",…}

- The set of possible transitions linking the characters of the affixes to the states A, q_I and q_F.

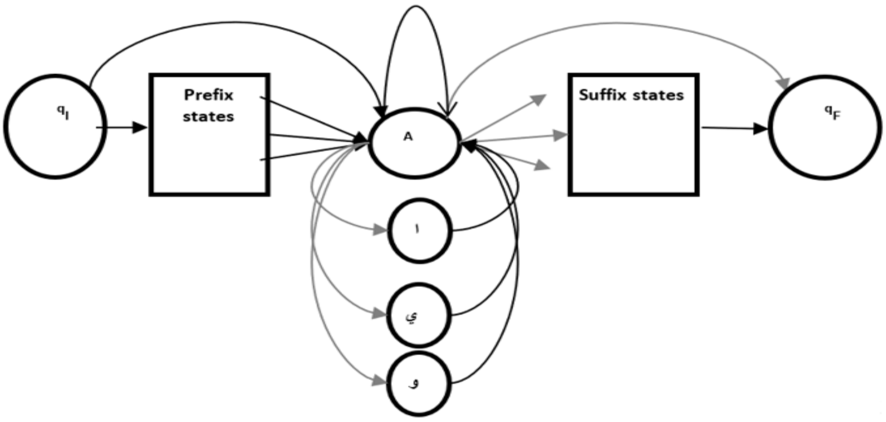

Figure 3. Diagram of our global network and possible transitions between states

How to find possible paths for the analysis of a word:

- To analyze a word W, we search the network for the different possible paths associated with W. These paths are given by:

  $$S = \left\{ \xi \in \mathrm{B} \,/ \mathrm{P_r}(\mathrm{w} \,/\, \xi) \neq 0 \right\}$$

  B: the possible paths in our network that have the same lengths as w.
- The solutions are the paths which make it possible to emit the word with a non-zero probability. we have adapted the Viterbi algorithm to the following format, to facilitate and reduce the calculation in (1):

  $$\delta_t(c_j) = \underset{c_i \to c_j}{\mathrm{NL}} \left(\delta_{t-1}(c_i) a_{ij} 1_{c\ j}(w_t)\right) \quad (1)$$

NL(x) is the non-zero value of x. We search for the states ci which give nonzero values of:

$$\delta_{t-1}(c_i) \cdot a_{ij} \cdot 1_{c_j}(w_t)$$

$\delta_T(q_F)$is the maximum probability of transmission of the word from a given path. By a recursive calculation we recover all the possible paths which give non-zero values (T the length of the word). With:

- a_{ij} : The state transition probability C$_i$ to C$_j$, where:

$$a_{ij} = \begin{cases} 1 \; if\ the\ transition\ is\ possible \\ 0 \; else \end{cases}$$

- w_t : T ththe character of the word w.

$$1_{c_j}(w_t) \;=\; \begin{cases} 1 & si \quad C_j{=}w_t \\ 0 & \text{sinon} \end{cases} \quad \text{We take} \quad 1_A(w_t){=}1$$

- Initialisation

$$\delta_0(c_i) = \begin{cases} 1 & si\ c_i{=}\ q_I \\ 0 & ifnot \end{cases}$$

The test was carried out on 20,000 words representing different grammatical categories (verbs and nouns). 96% of these words were correctly analyzed and our finite state automaton analyzer proposed different possible analyzes for these words, while it did not do so for the remaining 4%. 95% of these errors were due to not taking into account the calculation links between prefixes, roots and suffixes in our analyzer (Figure 4).

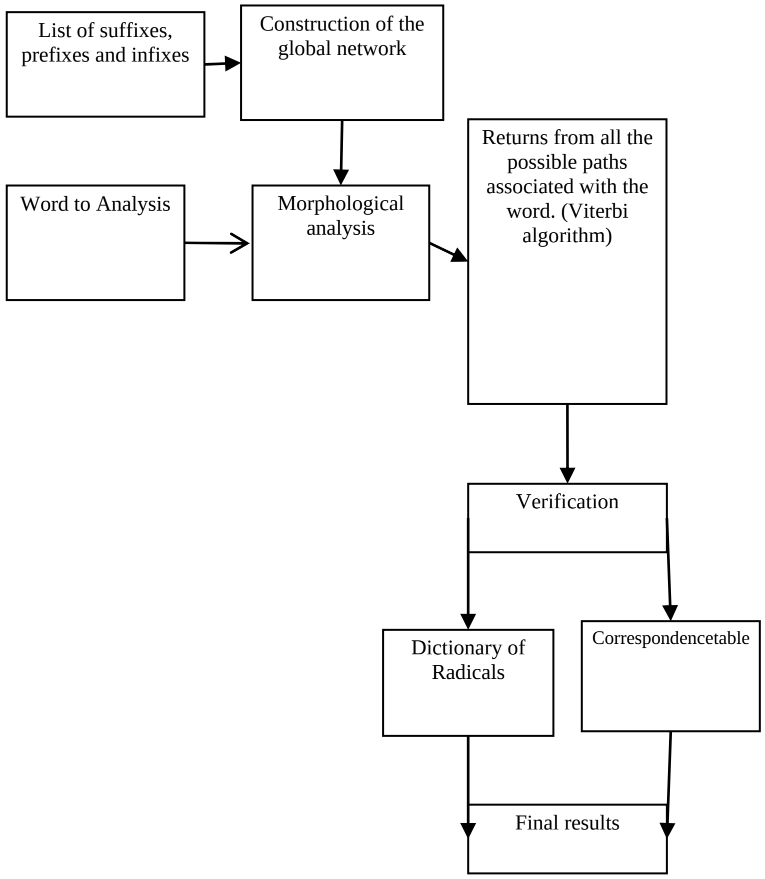
List of suffixes, prefixes and infixes
Construction of the global network
Word to Analysis
Morphological analysis
Returns from all the possible paths associated with the word. (Viterbi algorithm)
Verification
Dictionary of Radicals
Correspondencetable
Final results

Figure 4. Steps of our morphological analyzer based on finite state automata

To evaluate our approach based on finite state automata, we created the different dictionaries of suffixes, prefixes and radicals. After, we integrate the correspondence tables between affixes of the Arabic language. Then, from the list of prefixes, suffixes and infixes, we generate a global network of states as indicated previously,without using lexical dictionaries. This is the main advantage of our analyzer over Buckwalter arabic morphological analyzer and the analyzer based on finite state automata.

2.3. Integration of approaches to improve our analyzer

Our contribution is to develop a morphological analyzer of the Arabic language by integrating and combining approaches to improve our morphological analyzer. In this work, we have combined several approaches to improve a morphological analyzer that minimizes the error rate. For this, we have combined an approach that relies on the surface pattern with an approach that uses finite state automaton to analyze a given word.

We have many advantages in integrating and combining these approaches to develop and improve our analyzer. Among these advantages, we havereduction the size of the dictionaries used, unlike other analyzers. There is also the approach of the analyzer based on the surface patterns which uses the base of the surface patterns and models all the morphological variations of the derived words [20], [25]. On the other hand, the analyzer approach based on finite state automaton only articulates at the base of the roots. This approach, we generated a network of states without using lexical dictionaries. This is the main advantage of our analyzer over Buckwalter analyzer and surface pattern based analyzer.

In a finite state automaton based approach, the morphological analysis processes all the words, unlike the approach by patterns, which only gives the analyzes associated with the surface patterns existing in the database of patterns. The finite state automaton approach does not require the basics of linguistic knowledge to perform the morphological analysis of a word.However, these approaches have a few drawbacks. The surface pattern approach deals onlywith derived words, while the finite state automata approach process even non-derived words.

3. RESULTS AND DISCUSSION

The proposed adaptation comes from our study to a number of perspectives resulting from the words analyzed by our morphological analyzers of Arabic words. In this work, we have improved our morphological analyzer by integrating several approaches. These analyzers are essentially based on three concepts:

- The surface pattern [25];
- BuckwalterArabicmorphological analyzer [10] used with correspondence tables between Arabic language affixes;
- A finite state automaton [24].

A corpus of words from the Arabic language, which is developed by a group of linguists, is chosen as a grammatical support because it uses the same notation of the Arabic language, which is considered standard for natural language processing [17], [18].

The morphological analyzer has been implemented according to the structure of a network of finite state automaton. The database of surface patterns is made up of correspondence tables between affixes. A Viterbi algorithm was used to implement our network.

- The train dataset was 95% because it cover all categories, and test dataset was 5% of corpus which contains only the missed categories. As result, the morphological analyzer showed an accuracy of 95.09%.
- The morphological analyzer was trained on95% of the morphological corpus and tested on the remaining 5%. The test dataset is only 5% but it contains a significant volume of words which have never been used for training.

As result, the morphological analyzer showed an accuracy of 95.09%. The results obtained by our analyzer are comparable to those of analyzers in general, but lower than those of specialized Arab analyzers which reach an accuracy of 97.09% (Table 1). Nevertheless, there are several possibilities for improvement, because it is possible to increase the size of the test morphological corpusand include more contextual information to disambiguate.

Table 1. Accuracy of each morphological analyzer

Morphological analyzers	Our surface-pattern based analyzer	Our finite state automaton-based analyzer	Buckwalter Arabic morphological analyzer	Our adaptive approaches
Accuracy	94,41%	95,09%	93,87%	97,09%

4. CONCLUSION

This article presents a concept of finite state automaton and the integration of tables of correspondence between affixes in the analysis approach. Moreover, it shows the importance of using surface patterns, detailing its working mechanism and the main areas of application and the great importance of its use in the field of natural language processing. The effectiveness of our proposed approach has been demonstrated experimentally. Our morphology analyzer obtained results comparable to general analyzers, with indicators and lower than those presented by specialized Arabic language analyzers. Improvements will be acceptable, as increasing the size of the morphological corpus for disambiguation. Finally, we hope to improve the architecture of adaptive approaches to morphological analyzes of Arabic words, by incorporating mechanisms pour choisir les types modèles de calcul avec des critères d'évaluation bien réelle et des règles de transition claires.

REFERENCES

[1] N. H. Hegazi and El-Sharkawi, A.A, “Natural Arabic language processing,” in *Proceedings of the National Computer Conference*, 1986, pp. 10–17.

[2] K. R. Beesley, “Arabic morphology using only finite-state operations,” in *Proceedings of the Workshop on Computational Approaches to Semitic Languages - Semitic ’98*, 1998, p. 50, doi: 10.3115/1621753.1621763.

[3] Y. Hlal, “Morphological analysis of Arabic speech,” in *Workshop Papers Kuwait/Proceedings of Kuwait Conference on Computer Processing of the Arabic Language*, 1985.

[4] T. Buckwalter, Buckwalter Arabic Morphological Analyzer Version 2.0. *Linguistic Data Consortium, catalog number LDC2004L02 and ISBN 1-58563-324-0*, 2004.

[5] C. Gaubert, “Analyze morphology of a text by computer–Results and evaluation,” *AnIsl*, vol. 29, pp. 283–311, 1996.

[6] A. M. Azmi and R. S. Almajed, “A survey of automatic Arabic diacritization techniques,” *Natural Language Engineering*, vol. 21, no. 3, pp. 477–495, May 2015, doi: 10.1017/S1351324913000284.

[7] S. Khoja. and R. Garside, “Stemming Arabic text,” *Computing Department, Lancaster University, Lancaster, UK*, vol. 1, no. 1, pp. 1–20, 1999.

[8] I. A. Al-Sughaiyer and I. A. Al-Kharashi, “Arabic morphological analysis techniques: A comprehensive survey,” *Journal of the American Society for Information Science and Technology*, vol. 55, no. 3, pp. 189–213, Feb. 2004, doi: 10.1002/asi.10368.

[9] T. Buckwalter, “Buckwalter Arabic Morphological Analyzer Version 1.0,” *Linguistic Data Consortium, University of Pennsylvania, LDC Catalog No.: LDC2002L49*, 2002.

[10] J. Dichy, A. Farghaly, and U. Lumière-lyon, “Roots and patterns vs. stems plus grammar-lexis specifications: on what basis should a multilingual lexical database centred on Arabic be built ?,” in *Machine Translation*, 2003, pp. 1–8.

[11] S. Al-Fedaghi and F. Al-Anzi, “A new algorithm to generate Arabic root-pattern forms,” in *Proceedings of the 11th national Computer Conference and Exhibition*, 1989, no. January 1989, pp. 391–400.

[12] K. Beesley and L. Karttunen, “Finite state morphology homepage,” *Computational Linguistics*, vol. 30, no. 2, pp. 3–5, 2003, [Online]. Available: http://acl.ldc.upenn.edu/J/J04/J04-2006.pdf.

[13] M. Boudchiche, A. Mazroui, M. O. A. O. Bebah, A. Lakhouaja, and A. Boudlal, “AlKhalil Morpho Sys 2: A robust Arabic morpho-syntactic analyzer,” *Journal of King Saud University - Computer and Information Sciences*, vol. 29, no. 2, pp. 141–146, Apr. 2017, doi: 10.1016/j.jksuci.2016.05.002.

[14] S. Iazzi, A. Yousfi, M. Bellafkih, and D. Aboutajdine, “Morphological analyzer of Arabic words using the surface pattern,” *International Journal of Computer Science*, vol. 10, no. 2, pp. 254–258, 2013, [Online]. Available: http://ijcsi.org/papers/IJCSI-10-2-1-254-258.pdf.

[15] C. Audebert and A. Jaccarini, “From word recognition to speech (in France),” *Annales islamologiques*, vol. 24, 2016.

[16] K. R. Beesley, “Arabic finite-state morphological analysis and generation,” in *Proceedings of the 16th conference on Computational linguistics -*, 1996, vol. 1, p. 89, doi: 10.3115/992628.992647.

[17] T. A. El-Sadany and M. A. Hashish, “An Arabic morphological system,” *IBM Systems Journal*, vol. 28, no. 4, pp. 600–612, 2010, doi: 10.1147/sj.284.0600.

[18] K. Darwish, “Building a shallow Arabic morphological analyzer in one day,” in *Proceedings of the ACL-02 workshop on Computational approaches to semitic languages -*, 2002, pp. 1–8, doi: 10.3115/1118637.1118643.

[19] Y. Jaafar, K. Bouzoubaa, A. Yousfi, R. Tajmout, and H. Khamar, “Improving Arabic morphological analyzers benchmark,” *International Journal of Speech Technology*, vol. 19, no. 2, pp. 259–267, Jun. 2016, doi: 10.1007/s10772-016-9340-x.

[20] A. Yousf, “The morphological analysis of Arabic verbs by using the surface patterns,” *International Journal of Computer Science Issues*, vol. 7, no. 3, pp. 33–36, 2010.

[21] J. V. D. Hoek and R. J. Elliott, “The Viterbi Algorithm,” in *Introduction to Hidden Semi-Markov Models*, Cambridge University Press, 2018, pp. 56–63.

[22] J. Goldsmith, “Unsupervised learning of the morphology of a natural language,” *Computational Linguistics*, vol. 27, no. 2, pp. 153–198, Jun. 2001, doi: 10.1162/089120101750300490.

[23] A. Farghaly, “Arabic natural language processing: challenges and solutions,” *ACM Transactions on Asian Language Information Processing*, vol. 8, no. May 2013, pp. 1–10, 2003.

[24] K. Koskenniemi, “Two-level morphology: A general computational model for word-form recognition and production,” Thesis, University of Helsinki, 2005.

[25] A. Soudi, V. Cavalli-Sforza, and A. Jamari, “A computational lexeme-based treatment of Arabic morphology,” in *ACL 2001 Workshop on Data-Driven Machine Translation*, p. 8, 2001.

BIOGRAPHIES OF AUTHORS

Said Iazzi is a doctoral at Faculty of Science, Mohammed V University in Rabat, Discipline Engineering Sciences. His researches are in fields of computer sciences, automatic arabic language processing, AI, data science and machine learning, arabic handwriting recognition and correction of Arabic spelling errors. He can be contacted at email: iazzisaid@yahoo.fr.

Abderrazak Iazzi received the Master and Ph.D. degrees from Mohammed V University, Rabat, Morocco, in 2013 and 2021, respectively. His research interests include computer vision, deep learning, and intelligence data analysis. He can be contacted at email: abdou.iazzi@gmail.com.

Hicham Gueddah is an Assistant Professor researcher in Computer Science at Mohammed V University in Rabat-Morocco, currently joining the Intelligent Processing and Security of Systems Team-FSR at Mohammed V University. His field and scope of research is the natural language processing (NLP), text mining and deeplearning, particularly the research axis of automatic correction. He can be contacted at email: h.gueddah@um5r.ac.ma.

Abdellah Yousfi is a Professor at the Faculty of Law, Economics and Social Sciences Souissiat Mohamed V University in Rabat, since 2007. He is member of the Information, Communication, Embedded Systems and Natural language processing Team at the National High School of Computer Science and Systems Analysis (ENSIAS) Mohamed V University in Rabat, Morocco. His research interests include creation of corpora for the Arabic language, Arabic speech recognition, Arabic handwriting recognition and correction of Arabic spelling errors. He can be contacted at email: a.yousfi@um5r.ac.ma.

Mostafa Bellafkih received the Ph.D. thesis in Computer Science from the University of Paris 6, France, in June 1994 and Doctorat Es Science in Computer Science (option networks) from the University of Mohammed V in Rabat, Morocco, in May 2001. His research interests include the network management, knowledge management, AI, Data mining and database. He is Professor in The National Institute of Posts and Telecommunications (INPT) in Rabat, Morocco since 1995. M. His research in Computer Communications (Networks), Information Systems (Business Informatics) and Intelligent System. He can be contacted at email: bellafkih@inpt.ac.ma.