AL-FarahidiArabic Diacritizer

Labib Arafeh[1] and Iyad Abu Samrah[2]
[1] Faculty of Information Technology, Arab Open University
[2] Department of Information Technology, Ministry of Education and Higher Education, Palestine

Abstract: *The paper proposes an automatic Arabic Diacritizer namely, AL-Farahidi Arabic Diacritizer, AFAD. The proposed AFAD is based on a hybrid approach that combines the Statistical as well as the Morphological methods. The validation of AL-Farahidi Arabic Diacritizer is conducted making use of generated tens of sentences. Three metrics were used to check its performed adequacy. The achieved performance of AFAD include 92% Word Error Rate, 90% Diacritic Error Rate and 80% Sentence Error Rate. A performance comparison is accomplished between AFAD and the other available diacritizers namely, Mishkal, RDI and the MADAMIA. The performance of AL-Farahidi Arabic Diacritizer outperforms all other three diacritizers. Although, we have achieved these preliminary and promising results, it is still too early to declare its overwhelmed performance. As it requires further investigation and expansion such as testing the speed of processing, increasing the number of regular and irregular grammatical sentences, longer sentences, which will be the future work.*

Key Words: *Arabic Diacritizating Systems, MADAMIRA, Mishkal, RDI Standard Modern Arabic*

Received August20, 2017; Accepted November 8, 2017

1. Introduction

Arabic language belongs to the group of the Semitic alphabetical scripts that is based on 26 consonants,and optional diacritics to indicate vowels. The considered writing system is the Abjad writing, where there is one symbol per consonant, and the reader must provide the appropriate vowel to indicate inflectional or derived forms. Arabic Diacritics are glyphs (Harakat in Arabic) that are added to help clarify the meaning of words and clarify any vague spellings or pronunciations. Arabic diacritics usually include consonant pointing (iʿjām) and Tashkil. The Standard Modern Arabic is usually written with consonant pointing, whereas; Tashkil is mainly used in the Holy Qur'an, Hadiths, dictionaries, and pedagogical books for Arabic learners like kids as well as foreigners. In this paper, we will be referring to Tashkil as diacritics.Consonants consist of strokes and dots. Ten of them have one dot, three have two dots, two have three dots. Whereas, diacritics which are written as strokes and can change the pronunciation and the meaning of the word. There are three types of diacritics. One type may appear as strokes above the character such asFatha, Dhamma, Sukun, Shadda or Maddah. This type occurs at either the beginning or middle or end of the word. The second type may appear also at the beginning or middle or end, but belowthe character like Kasra. The third type is known as Tanween that occurs only at the end of the word, which is a form of discretizing Arabic writing with double Fatha, double Dhamma or double Kasra. Tanween.

Nowadays, we are witnessing the advancement of the Text-to-Speech (TTS) systems that require an embeddeddiacritization algorithm to the Natural Language Processing (NLP) system [11]. They particularly require discourse analysis, Part-of Speech-Tagging, Named Entity Recognition, Sentence Breaking as well as Word Sense Disambiguation. However, for Arabic learners and automatic computer processing, this is not possible. Thus, a diacritizer tool is an essential step to mimic the human capability to identify the proper vocalization of the text.

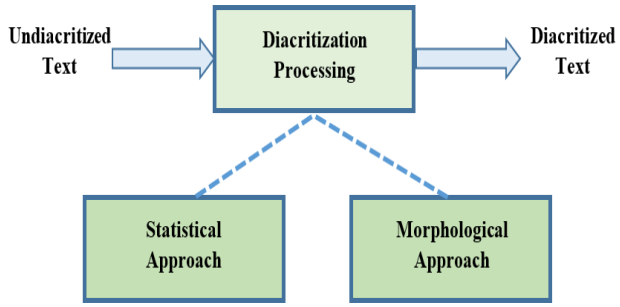

Figure 1. Al-Farahidi Arabic Diacritizer Block Diagram

Research on Arabic computational morphology has increased considerably in recent years. Indeed, research on Arabic morphology has always been not extraordinarily prolific due to the complexity of the topic. However, despite the reasonable number of computational models that have been proposed, the different approaches have not been completely explored and a vast amount of continued work is still needed [21].

A prototype of an Arabic Diacritization system, which is called as Al-Farahidi Arabic Diacritizer (AFD) is presented as illustrated in figure 1. The proposed hybrid diacritization AFAD is based on the statistical and the morphological approaches. AFAD accepts Undiacritized texts, starts diacritizating the text with the morphological approach. If done, it produces the diacritized text, otherwise it further diacritizating the text with the statistical approach and produces the diacritized text. The performance of the proposed AFAD is evaluated against three well known developed diacritization systems including Mishkal [6], RDI [18]and the MADIMAR [19].

The paper is composed of five sections as follow: Section two provides a literature survey to briefly reveal the latest advances and approaches in the field. While section three describes the hybrid approach of Al-Farahidi Arabic Diacritizer, section four presents the performance of Al-Farahidi Arabic Diacritizer in comparison with its counterparts' approaches. Section five discusses the obtained results, whereas, section six concludes the paper and highlights the future works.

2. Literature Review

Several researchers have addressed the diacritization problem. These include, the process of building an Arabic speech corpus aiming at collecting large amounts of Arabic speech data that consists of 51 thousand words [11]. An approach proposed by[1]integrates a wide array of lexical, segment-based andpart-of-speechtag features. Authors reported a diacritic error rate of 5.5%, a segment error rate 8.5%, and a word error rate of 17.3%. In case-ending-less setting, they achieved a diacritic error rate of 2.2%, a segment error rate 4.0%, and a word error rate of 7.2%. Mohsen and coauthors [7] introduced a two-layer stochastic system to automatically diacritize raw Arabic text. The first layer tries to decide about the most likely diacritics by choosing the sequence of full-form Arabic word diacritizations with maximum marginal probability via long A* lattice search and m-gram probability estimation. When full-form words happen to be out-of-vocabulary, the second layer is resorted to. This second layer factorizes each Arabic word into its possible morphological constituents (prefix, root, pattern and suffix), then uses m-gram probability estimation and A * lattice search to select among the possible factorizations to get the most likely diacritizations sequence. They reported an 11.5% morphological error in factorization diacritizer and 9.2% in hybrid diacritization. With regards to syntactical errors, they found it to be 26.1% for factorizing diacritizers and 21% for hybrid diacritization, when the using 128K training corpus size. A novel approach has been developed by Habash and coauthors [5]a diacritization system for written Arabic which is based on a lexical resource. It combines a tagger and a lexeme language model. They reported a word error rate of 16% and a diacritic error rate of 5.3%. Rani and coauthors presented an algorithm for restoring these symbols using cascade probabilistic finite state transducers on the Arabic tree bank by integrating a word based language model, a letter-based language model, and morphological model [17]. Their model was expressed as a finite state model based on the Viterbi decoding and consists of several transducers like: language model, spelling, diacritic drop and unknowns. They achieved 15.48% and 30.39%-word error rates, and 17.33% and 24.03% diacritization error rates for without the case and with the case, respectively. Several other researchers who contributed to the progress of Arabic diacritization include: [20]who treated diacritization as a Machine Translation Problem and as a Sequence Labeling Problem research; [16] who studied the Arabic Diacritization in the Context of Statistical Machine; [13]who developed a statistical-based Automatic Restoration of Arabic Diacritics; [10]who used statistical approach for language modeling; [12] developed a Xerox Finite-State Technology-based Xerox Arabic Morphological Analyzer and Generator; [7]developed a morphological analyzer and generator for MSA and the spoken dialects, called MAGEAD;[14]presented a MADA+TOKAN toolkit that includes different NLP tools for Arabic language processing; [15] developed a large-scale finite-state morphological analyzer toolkit, known as the AraComLex; [3] developed an Arabic stemmer that removes any affixes from words and reduces these words to their roots; [8]developed a morphological analyzer that identifies vowelizations, proclitic and enclitics, nature of the word, voweled patterns, Stems, Roots and syntactic form; and the Hidden Markov Model-based automatic morphological annotation tool, which is called PurePos [9] that can perform tagging and lemmatization at the same time.

Furthermore, current diacritizers include: The Mishkal [6] tool which, can be accessed either online or offline; the RDI [18] tool which, is based on the morphological and syntactical diacritization methods; and the MADAMIRA [21] tool that combines the MADA [5] and AMIRA [16] tools to perform morphological analysis and disambiguation of Arabic.

Commercially wise, the most currently available industrial Arabic morphological processors include Sakhr's [2], Xerox's [12] and RDI's[18].Sakhr'stool which, is a factorizing one based on the standard Arabic dictionaries,declares 97% accuracy. Xerox's system is also a factorizing system based on the standard Arabic dictionaries. RDI's system which

declares a 96% accuracy, is a factorizing system where each regular derivative root is allowed to combine with any form as long as this combination is morphologically allowed.

3. Al-FarahidiArabic Diacritizer

The proposed AL-FarahidiArabic Diacritizerreads an undiacritized text which, is provided by user, diacritizes it and outputs the corresponding diacritized text of each input word as well as other attributes like: suffix, prefix, root, and the grammar (verb, noun, letter, etc.). AFAD is composed of several major components including the Reading Texts, Database, Search engine, the Statistical Analysis, and Morphological Analysis and the outputting component. Furthermore, AFAD implements both the statistical and the morphological analysis approaches. The following subsections briefly describes all of these components.

The architecture of the proposed hybrid AFAD is described in figure 2.

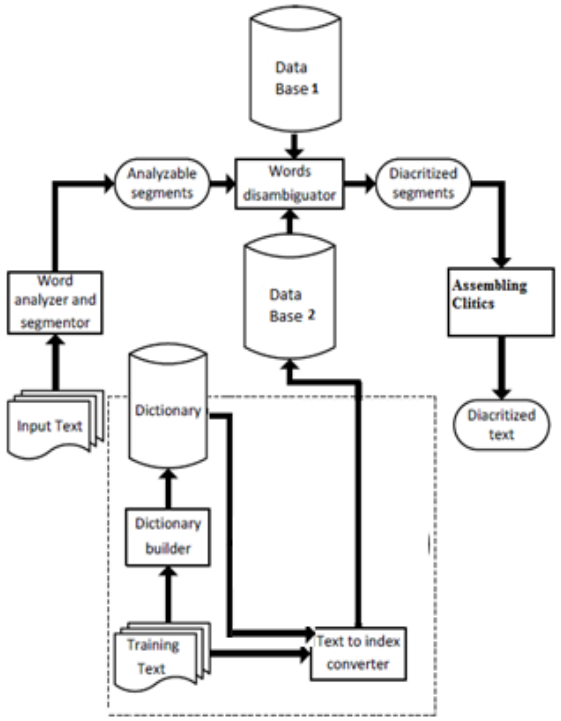

Figure 2. The Hybrid AL-FarahidiArabicDiacritizerArchitecture

The data flow of figure 2 is described by the following algorithm for a single text:

Start

- *ReadUndiacritized text;*
- *Preprocess the undiacritized text by dividing the sentence into words;*
- *Perform diacritization process:*
  - *Perform morphological test (including Analyzing, Isolation, Lookup at Closed Lists, Un-diacritized Pattern Matching and Root Extraction)*
  - *Check available DB 1*
    - *If found, produce diacritized text*
    - *Else,perform statistical test*
      - *If found produce diacritized test*
      - *Else user suggestion*
    - *Update DB 1*
- *Produce the corresponding diacritized text*

Stop.

The statistical approach relies heavily on the training data are available system. Sub-model statistical language consists of three main steps.

1. Create a list of commonly used phrases in Arabic diacritized well.
2. Create a copy diacritized is to build a training model.
3. Use the list created in Step 1 to diacritize Arabic text.

Themorpho-syntactical approach uses AlKhalilMorpho System.Morphology studies the internal structure of words. That is, it analyzes the structure of morphemes and other units of meaning in a language. The four morphological processes are: Derivation that produces nous (nous in Arabic includes adverbs, adjectives, pronouns, proper nouns and many others) and verbs from the roots (first stem of verbs). So, the roots, which are verbs consist of three (most cases), four and five letters (rare roots), are the origin of all the Arabic words; Inflection that is produced by adding some well-known affixes (prefixes, suffixes and infixes) in order to give some attributes to the word; Cliticization that uses Clitics (Clitic: a word that is written or pronounced as part of another word); and Compounding that combines two words.

4. Al-FarahidiArabic Diacritizer Validation

The proposed AFAD system has been validated with forty undiacritized sentences that cover ten different grammatical rules. These sentences are fed one by one to the AFAD making use of the user interface as shown in figure 3.

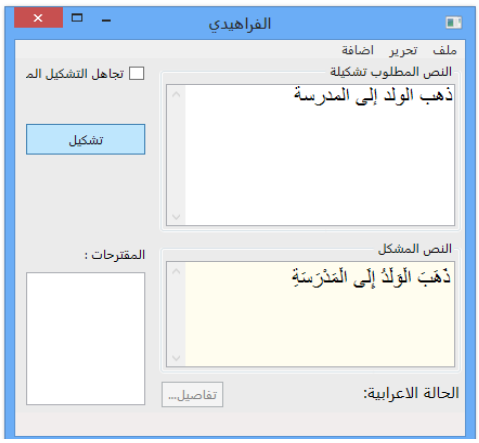

Figure 3. Al-Farahidi Arabic Diacritizer's User Interface

Furthermore, the same undiacritized sentences were tested (diacritized) using three well-known Arabic Diacritizers namely, Al-Mishkal [6] RDI [18]and the MADAMIRA [21].

We have used three different metrics to measure the adequacy of the AFAD tool as well to compare the performance of the three diacritizers against that of the AFD. These metrics include:

1. The Word Error Rate (WER) which, is thepercentage of words that were incorrectly diacritized; WER is given byequation 1.

$$WER = \left(\frac{\sum \frac{Number\ of\ Corrected\ Words}{Number\ of\ Words}}{Total\ Number\ of\ Sentences} \right) \times 100\% \quad (1)$$

2. The Diacritic Error Rate (DER), which is the percentage of diacritics, including the null diacritic, that were incorrectly predicted); DER is given by equation. (2).

$$DER = \left(\frac{\sum \frac{Number\ of\ Corrected\ Diactrics}{Number\ of\ Diactrics}}{Total\ Number\ of\ Sentences} \right) \times 100 \quad (2)$$

3. Sentence Error Rate (SER), which is given by equation (3).

$$SER = \left(\frac{Number\ of\ Corrected\ Sentences}{Total\ Number\ of\ Sentences} \right) \times 100\% \quad (3)$$

5. Results and Discussion

The obtained results are tabulated in table 1 as follow:

Table 1.Arabic Diacritizers Performance.

Metrics / Diacritizers	WER	DER	SER
Mishkal	60%	60%	32.5%
RDI	6.4%	69%	0%
MADAMIRA	62%	80%	35%
AL-FARAHIDI	92%	90%	82%

Table1 shows the result of the error measures WER, DER and SER for the four diacritizers.

Concerning the WER metric, Al-Farahidi tool scores 90% follow by almost equal performance of around 60% for Mishkal and MADAMIRA tools.Whereas, RDI scores around 6%.systems. It might be that the validated sentences were not included in their corpus. In addition, it can be noticed that the MADAMIRA and MISHKAL have performed almost equally but less than Al-Farahidi tool as they do not concentrate at the end of the word.

Regarding the DER metric, Al-FarahidiArabic diacritizeroutperforms all other diacritizers with 90% score. The MADAMIRA tool scores 80 % whereas; RDI and Mishkal score 69% and 60%, respectively. TheDER measure reflects the diacritized dialects. This shows that other tools do not focus on the end of the word dialect.

Considering the SER metric, Al-FarahidiArabic diacritizerbeats all other diacritizers.WhileAl-FarahidiArabic diacritizer scores 82%, the MADAMIRA, Mishkal and RDI score 35%, 32.5% and 0%, respectively. For the RDI tool, as it does not support the end of word diacritization, the score is 0%.

Thus, the hybrid approach of morphological and statistical methods that is embedded in the Al-FarahidiArabic Diacritizer performs much better than its three counter parts tools.

6. Conclusion and Future Work

The paper briefly reviews the various available Arabic diacritization approaches and tools. A hybrid-based Arabic diacritizeris proposed called as AL-FarahidiArabic Diacritizer. It combines both the Statistical as well as the Morphological methods. The validation of AL-FarahidiArabic Diacritizer is conducted making use of generated tens of sentences covering ten different grammatical rules. Its performed adequacy is measured using the metrics of Word Error Rate, Diacritic Error Rate and the Sentence Error Rate. A performance comparison is accomplished between AL-FarahidiArabic Diacritizer and the other available diacritizers namely, Mishkal, RDI and the MADAMIA. The performance of AL-FarahidiArabic Diacritizer outperforms all other three diacritizers.

Although, we have achieved these preliminary and promising results, it is still too early to declare overwhelmed performance. As it is required to check the speed of processing, more regular and irregular grammatical sentences, longer sentences, which will be the future work.

References

[1] Alghamdi, M. et al, "Automatic restoration of Arabic diacritics: A simple, purely statistical approach. "Arabian Journal for Science and Engineering", 35.2, 2010.

[2] Almosallam, I.; Alkhalifa, A.; Alghamdi, M.; Alkanhal, M.; Alkhairy, A.: SASSC: A standard Arabic single speaker corpus. In: 8th ISCA Synthesis Workshop, Barcelona, Spain, pp. 249–253, 2013.

[3] Alghamdi, M. et al. "A hybrid system for automatic Arabic diacritization, "*The 2nd International Conference on Arabic Language Resources and Tools",* 2009.

[4] AlkhalilMorpho Sys, Available at http://alkhalil-morpho-sys.soft112.com/ (Accessed: 15 December 2015).

[5] Angelova, G. et al, Recent Advances in Natural Language Processing, RANLP 2013, 9-11 September, 2013, Hissar, Bulgaria. RANLP 2011 Organizing Committee / ACL2013, 2013.

[6] ArabDiac: RDI's Automatic Arabic Phonetic Transcriptor (Diacritizer/Vowelizer), available at http://www.rdi-eg.com/technologies/arabic_nlp.htm#ArabDiac(Accessed: 15 December 2015).

[7] Arabic Language Disambiguation for Natural Language Processing Applications, Available at http://innovation.columbia.edu/technologies/cu14012_arabic-language-disambiguation-for-natural-language-processing-applications (Accessed: 15, December 2015.

[8] Arabic stemmer, Available at: http://zeus.cs.pacificu.edu/shereen/research.htm (Accessed: 15 December 2015).

[9] Arfath, P., et al, "MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic" (2014), http://www.lrec-conf.org/proceedings/lrec2014/pdf/593_Paper.pdf, Accessed on Jan. 2016.

[10] Attia, M., et al. "An open-source finite state morphological transducer for modern standard Arabic, "Proceedings of the 9th International Workshop on Finite State Methods and Natural Language Processing. Association for Computational Linguistics, 2011.

[11] Beesley, K. R., Finite-State Morphological Analysis and Generation of Arabic at Xerox, Available at: https://www.researchgate.net/publication/2373183_Finite-State_Morphological_Analysis_and_Generation_of_Arabic_at_Xerox_Research_Status_and_Plans_in_2001, (Accessed: 15 December 2015).

[12] Diab, M., et al, "Arabic diacritization in the context of statistical machine translation. "*Proceedings of MT-Summit*, 2007.

[13] Elshafei, M., et al, "Statistical methods for automatic diacritization of Arabic text", The Saudi 18th National Computer Conference. Riyadh. Vol. 18. 2006.

[14] Habash, N. and Owen, R., "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*. Association for Computational Linguistics, 2007.

[15] Habash, N., and Owen, R., "MAGEAD: a morphological analyzer and generator for the Arabic dialects. "Proceedings of the 21st International Conference on Computational Linguistics *and the 44th annual meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2006.

[16] Habash, N., et al. "MADA+ TOKAN: A toolkit for Arabic tokenization, diacritization, morphological disambiguation, pos tagging, stemming and lemmatization. "Proceedings of the 2nd International Conference on Arabic Language Resources and Tools (MEDAR), Cairo, Egypt, 2009.

[17] Mishkal: Arabic Text Vocalization, available at http://sourceforge.net/projects/mishkal/(Accessed : 15 December 2015.

[18] Nelken, R., et al. "Arabic Diacritization using weighted finite-state transducers." *Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages*. Association for Computational Linguistics, 2005.

[19] Schlippe, T., et al. "Diacritization as a machine translation problem and as a sequence labeling problem. "*Proceedings of the Eighth Conference of the Association for Machine Translation in the Americas (AMTA), Hawai'i, USA*. 2008.

[20] SHAKKIL: An Automatic Diacritization System for Modern Standard Arabic Texts, Available at: http://aramedia.com/diacritizer.htm (Accessed: 14 January 2016).

[21] Zitouni, I, Sorensen, J. S. &Sarikaya, R. "Maximum Entropy Based Restoration of Arabic Diacritics", Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (ACL); Workshop on Computational Approaches to Semitic Languages; Sydney, Australia, 2006.

LabibArafeh has over Thirty years of professional experience including Information Technology applications, teaching, training, administration, development and planning, tied with hands-on exposure monitoring, evaluation & supervisory responsibilities. Dr. Arafeh has obtained his expertise from working experience at three universities, study visits and as the director of the Palestinian Accreditation & Quality Assurance Commission, as well leading and participating in developing and implementing several local and global related projects. He has also been involved in managing, supervising and implementing several international & local projects such as developing e-Learning and quality assurance policies for EMUNI University. Dr. Arafeh has been involved in leading and participating in related several international & local projects including the UNESCO funded IT & Electrical Engineering Benchmarks, RAND (US)- Al-

Quds University funded effective teaching project, EU-supported FINSI, ICT-LEAP and RUFO, Jamila Tempus projects. In addition, Dr. Arafeh has participated in evaluating & studying positive and negative impacts of several World Bank & EU proposed projects. In addition, he evaluated the technology 5-10 school curricula and developed the textbook for the 8th grade. Furthermore, Dr. Arafeh's research interests lie in major aspects of computing applications for people, including Essay-Type Auto Grading Systems; development of e-learning systems for practical / experimental courses; Applying Augmented, Virtual and Mixed Realties in various applications including Education; Quality Assurance and evaluation of e-learning systems and websites; applications of soft computing techniques in forecasting, predictions and political conflicts; promoting the use of technology in teaching, and the production of educational and cultural multimedia, as well as Human Security.

Iyad Abu Samrah obtained his B.Sc. and M.Sc. Engineering degrees from Al-Quds University, Palestine. Currently, Mr. Abu Samrah is a senior member at the IT department, Ministry of Education and Higher Education, Palestine. His research areas include Education-based on Mixed Reality Applications, Text-to-Speech Development, mainly for disables, Graphic Design and Web Evaluation, Databases and Networks.