International Journal of Computing & Information Science
Automatic Diacritics Restoration for Dialectal Arabic Text
Ayman A. Zayyan, Mohamed Elmahdy, Husniza binti Husni, Jihad M. Al Ja'am
Pages 159 – 165
DOI: http://dx.doi.org/10.21700/ijcis.2016.119

Automatic Diacritics Restoration for Dialectal Arabic Text

Ayman A. Zayyan[*], Mohamed Elmahdy[‡], Husniza binti Husni[†], Jihad M. Al Ja'am[*]

[*]Department of Computer Science and Engineering, Qatar University, Doha, Qatar

[‡]Faulty of Media Engineering and Technology, German University in Cairo, Cairo, Egypt
[†]School of Computing, College of Arts and Sciences, Universiti Utara Malaysia, Malaysia
zayyan@qu.edu.qa, mohamed.elmahdy@guc.edu.eg, husniza@uum.edu.my, jaam@qu.edu.qa

Abstract: *In this paper, the problem of missing diacritic marks in most of dialectal Arabic written resources is addressed. Our aim is to implement a scalable and extensible platform for automatically retrieving the diacritic marks for undiacritized dialectal Arabic texts. Different rule-based and statistical techniques are proposed. These include: maximum likelihood estimate, and statistical n-gram models. The proposed platform includes helper tools for text pre-processing and encoding conversion. Diacritization accuracy of each technique is evaluated in terms of Diacritic Error Rate (DER) and Word Error Rate (WER). The approach trains several n-gram models on different lexical units. A data pool of both Modern Standard Arabic (MSA) data along with Dialectal Arabic data was used to train the models.*

Keywords: *Diacritization; Vowelization; Sialectal Arabic; Text Processing*

Received: June 04, 2016 | ***Revised:*** August 10, 2016 | ***Accepted:*** August 25, 2016

1. Introduction

Arabic is the largest Semitic language that is still in use today in terms of number of speakers, that exceeds 300 million. Arabic is spoken natively by people in 22 countries. It is also used in religious texts, such as the Holy book of AL Qur'an. Around 1.6 billion Muslims in the world are required to use Arabic verses in their daily prayers. However, many different Arabic dialects are in use in various different countries. Arabic dialects are used as the daily life communication language between the people. Moreover, social media users over the Arab world tend to use dialectal Arabic rather than MSA. Arabic speakers are able to understand the MSA that is used today in official communication and media. However, Arabic text, standard and dialectal, is currently written without diacritization or vowels (known as Tashkeel) which often makes the meaning ambiguous and semantically inaccurate, especially for non-native Arabic speakers. Arabic books and daily newspapers are written without diacritic marks.

Therefore, natural language processing tasks, such as automatic text-to-speech tasks, translations, and Arabic text mining (retrieve the exact words in the queries) may produce false results for undiacritized Arabic text. For instance, the Arabic word (علم) written without diacritic marks can have completely different meanings. In fact, it could be translated as the words "flag," "teach," "taught," "understood," or "science". The same is true for the word كتب. It can mean كَتَبَ "wrote" (Kataba), كُتِبَ "was written" (Kutiba), كُتُب "books" (Kutub), كَتَّبَ "made someone write" (Kattaba), or كُتِّبَ "was forced to write" (Kuttiba).

Special diacritic marks are also used in Arabic text to show vowel absence or consonant doubling. Restoring the diacritics to the text simplifies its pronunciation and proper understanding. Native Arabic speakers can mentally predict the correct meaning of the word based on the context. Automatic diacritics restoration, also known as vowelization, is a challenging task but it is necessary for most Arabic natural-language processing and computational-linguistic tasks. Manual diacritics restoration is a tedious and impractical solution. This task would require many experts (rendering the solution expensive), and much time would be required. It has been stated that, on average, one expert is able to revise

a mere 1,500 words on the results of an automatic diacritizer per day. It is also impractical for real-time tasks. For instance, in the case of a text-to-speech engine, it would be impossible to manually add diacritics to the amount of data required to read aloud websites for visually impaired people.

These problems with the manual approach have created a need for an automated diacritic restoration tool. Tools for automatic diacritization have been in development since the late 1980s. This has since been a continuously active research field, with many methods being implemented to tackle the problem. Several approaches have been proposed in literatures [1][2]. Linguistics rules, morphological analysis [3], statistical modeling, artificial neural networks [4], maximum entropy [5], hidden Markov model, n-gram language models, finite state automata, dynamic programming, statistical machine translation, semi-automated restoration [6], stochastic-based [7] and the Viterbi Algorithm have all been used.

Most prior methods, despite their complexity, still fall short of the desired outcome, that of a near perfect diacritic restoration. Moreover, they tend to deal with one, or very limited number of text genres. Most reported methods are known to have a Diacritization Word Error Rate (WER) between 3.5% and 17%. The field is thus still in active research and needs much work, both in the proposing of new approaches, and enhancing of old ones. In this paper, the problem and the recent advancement of Automatic Arabic text diacritization in addition to the available systems like FASSIEH [6], and ArabDiac, Al-Alamia, CIMOS, KACST [8] will be discussed. Next, a novel approach will be proposed to tackle this problem.

Most of prior work, as shown later in the next section, has focused on MSA. The aim of this paper is to build an automatic diacritization system for dialectal Arabic rather than MSA. There exist significant lexical, morphological, and syntactic differences between MSA and dialectal Arabic to the extent to consider them completely different languages [9].

A major problem that is specifically challenging for most of natural language processing tasks, that are related to dialectal Arabic, is the very limited available labeled data.

In this paper, we will rely on a small amount of vowelized dialectal Arabic Data, and we will augment the trained dialectal models with existing relatively larger amounts of MSA data. Egyptian dialectal Arabic has been chosen as a typical Arabic dialect.

2. Related Work

Several techniques have been used to tackle the Arabic diacritization problem. The techniques used in this area are mainly divided into three approaches: rule-based approaches, statistical approaches, and hybrid approaches.

2.1 Rule-based approach: A tagging system was proposed that classifies the words in a non-vocalized Arabic text to their tags. The system goes through three analysis levels. The first level is a lexical analyzer, the second level is a morphological analyzer, and the last level is a syntax analyzer. The system performance was tested using a data set with a total of 2,355 non-vocalized words selected randomly from newspaper articles. The reported accuracy of the system was 94% [10]. A rule-based diacritization system for written Arabic was presented; this system based on a lexical resource, which combines a lexeme language and tagger model. They used ATB3-Train with total of 288,000 words for training purpose and ATB3-Devtest with total of 52,000 words for testing purpose. The best result reported by their system was 14.9% WER and 4.8% DER. The authors also considered the case ending, and their system reported 5.5% WER and 2.2% DER [11].

2.2 Statistical approach: The new statistical approach proposed for Arabic diacritics restoration is based on two main models. The first is a bi-gram-based model which handles vocalization. The second is a 4-gram letter-based model to handle the OOV words. The authors used a corpus retrieved automatically from the Al-Islam website[1]. This corpus is an Islamic religious corpus containing a number of vocalized subjects (Quran Commentaries, Hadith, etc.). A vocalized Holy Qur'an was also downloaded from the Tanzil website[2] and merged with the corpus. Training set to testing set ratio was 90% to 10% respectively. The system varied in its report of WER from 11.53% to 16.87% based on the applied smoothing model. DER varied from 4.30% to 8.10% based on the applied smoothing model. Case-ending was considered in their research and their system reported WER varying from 6.28% to 9.49% based on the applied smoothing model, and DER varying from 3.18% to 6.86% based on the applied smoothing model [12].

A statistical approach proposed an automatic diacritization of MSA and Algiers dialectal texts. This approach is based on statistical-machine translation. The authors first investigate this approach on MSA texts using several data sources and extrapolated the results on

[1] http://www.al-islam.com/

[2] http://tanzil.net/

available dialectal texts. For MSA corpus, they used Tashkeela[3], a free corpus under GPL license. This corpus is a collection of classical Arabic books downloaded from an on-line library. It consists of more than 6 million words. They split the data on a training set of 80%, developing set of 10%, and a testing set of 10%. For comparison purposes, they used the LDC Arabic Treebank (Part3, V1.0). For a dialect corpus, Algiers corpus was manually developed. Initially it did not contain diacritics and it was vocalized by hand. The vocalized corpus consists of 4,000 pairs of sentences, with 23,000 words. For MSA, WER reported by their system was 16.2% and 23.1% based on the corpus in use, while DER reported was 4.1% and 5.7% based on the corpus in use. For Algiers dialect corpus, WER reported by their system was 25.8%, DER reported by their system was 12.8% [13].

Another algorithm has been proposed that relies on a dynamic programming approach. The possible word sequences with diacritics are assigned scores using statistical n-gram language modeling approach and different smoothing techniques used in this research, such as: Katz smoothing, Absolute Discounting and Kneser-Ney for Arabic diacritics restoration. For training and testing purposes, the authors used the Arabic vocalized text corpus Tashkeela. The corpus is free and collected from the Internet using automatic web crawling methods. It contains 54,402,229 words. The author divided the corpus into training and testing sets. The training set consisted of 52,500,084 word, while the testing set consisted of 1,902,145 words, which means 96.5% of the corpus was used for training purposes, and 3.5% was used for testing purposes. The WER for this system varied from 8.9% to 9.5% depending on the applied smoothing model. The WER considering the case-ending varied from 3.4% to 3.7% based on the applied smoothing model [14].

A new search algorithm was developed that supports higher order n-gram language models. The search algorithm depends on dynamic lattices where the scores of different paths are computed on the run time. For training and testing purposes, the authors used the Arabic vocalized text corpus Tashkeela. The authors divided the corpus into training and testing sets. The training set consisted of 52,500,084 words, while the testing set consisted of 1,902,145 words, which means 96.5% of the corpus was used for training purposes, and 3.5% was used for testing purposes. The WER for this system varied from 8.9% to 9.2% based on the applied model. The WER considering the case-ending varied from 3.4% to 3.6% based on the applied model [15].

The Empirical study proposed using different smoothing techniques commonly used in speech recognition and machine translation fields. The WER for this system varied from 8.9% to 9.5% based on the applied smoothing model. The WER considering the case-ending varied from 3.4% to 3.7% based on the applied smoothing model [16].

2.3 Hybrid approach: A different technique has been proposed that is based on deep learning framework, which includes the Confused Sub-set Resolution (CSR) method to improve classification accuracy, in addition to an Arabic Part-of-Speech (PoS) tagging framework using deep neural nets. The authors used TRN_DB_I and TRN_DB_II for training purposes, with a 750,000 word data-set and 2,500,000 word data-set respectively which was collected from different sources and diacritized manually by expert linguists. For testing purposes, TST_DB was used with an 11,000 word test set. Their system reported syntactical accuracy varying from 88.2% to 88.4% based on the data-set in use, and 97% morphological accuracy [17].

In [18], an approach based on a sequence transcription was proposed. A recurrent neural network is trained to recover the diacritic marks of undiacritized Arabic text. The authors used a deep bi-directional long short-term memory network that builds high-level linguistic abstractions of text and exploits long-range context in both input directions. The authors used data from the books of Islamic religious heritage, along with the Holy Qur'an. These 11 books are written with full diacritization marks. 88% of the corpus was used for training and the remaining 12% for testing. The WER in their system varied from 5.82% to 15.29% based on the used data. The DER varied from 2.09% to 4.71% based on the used data. They considered the case-ending, and the WER ranged from 3.54% to 10.23%, whilst the DER ranged from 1.28% to 3.07%.

3. Dataset

In this paper a high-quality diacritized Arabic corpus for MSA in addition to one dialectal Arabic corpus have been used, namely the standard LDC Arabic Tree Bank dataset [19] and the CallHome Arabic corpus of telephone speech [20]. The standard LDC Arabic Tree Bank is an Arabic text vocalized corpus, consists of 600 documents (≈340K words) from AnNahar newspaper. The corpus consist of different articles, including those from broadcast news, business, general news, interviews, Islamic topics, legal political debate, political news, scientific press and sports press.

The CallHome Arabic corpus of telephone speech was collected and transcribed by the Linguistic Data Consortium primarily in support of the project on Large Vocabulary Conversational Speech Recognition

[3] http://sourceforge.net/projects/tashkeela/

(LVCSR), sponsored by the U.S. Department of Defense. This release of the CallHome Arabic corpus consists of 120 unscripted telephone conversations between native speakers of Egyptian Colloquial Arabic (ECA), the spoken variety of Arabic found in Egypt. The dialect of ECA that this corpus represents is Cairene Arabic. The transcripts cover a contiguous 5 or 10 minute segment taken from a recorded conversation lasting up to 30 minutes. The data-set has been divided into a training set of 90% and a testing set of 10%.

4. Methodology

A Multi-lexical levels statistical based approach is used to estimate missing diacritic marks for a given un-vowelized Arabic text. The proposed approach can be configured to run on two different contextual lexical levels. The lexical levels are: word-level and letter-level. This approach has been applied and tested on two high-quality diacritized MSA Arabic corpora, namely Nemlar written corpus and Le Monde Diplomatic corpus. The system resulted in WER and DER of 5.1% and 2.7% respectively [21].

Word-Level Lexical Modeling: In this level, four different statistical n-gram models are adopted to re-introduce the missing diacritization marks:

(a) Four-gram Language Models: This model is adopted for use in gathering contextual information for adding diacritics of certain word. To improve diacritization accuracy, the results of this model are split into two sub models:

(a.1) Right-context Four-gram Model: In this model history (right context) of the given word will be considered to re-introduce the diacritization marks for the given word. More formally, each un-diacritized word in the test set will be replaced by the maximum likelihood estimate that corresponds to the diacritized word that occurred most frequently in the training set. This is done by considering the previous history of that word.

In this case the diacritizer will choose $word_i^d$ as the diacritized form of the input word represented by $word_i^u$ considering the previous history of that word represented by $word_{i-1}^u$, $word_{i-2}^u$ and $word_{i-3}^u$, as per the following equation:

$$word_i^d = \text{argmax p}\left(word_i^d \middle| word_i^u, word_{i-1}^u, word_{i-2}^u, word_{i-3}^u\right)$$

Where $word_i^d$ represents the selected diacritized form of the i^{th} un-diacritized word represented by $word_i^u$, given the previous history of the word represented by $word_{i-1}^u$, $word_{i-2}^u$ and $word_{i-3}^u$. In case the word was not found to have a right-context 4-gram, the system backs off to a left-context 4-gram model.

(a.2) Left-context Four-gram Model: As a further measure to improve the word level accuracy and diacritization level accuracy, a left-context model is built in a similar way to what is done in the previous sub model. However word left-context is considered instead of right context.

In this case, the diacritizer will choose $word_i^d$ as the diacritized form of the input word $word_i^u$ and the words next to the given one to be diacritized $word_{i+1}^u$, $word_{i+2}^u$ and $word_{i+3}^u$ as per the following equation:

$$word_i^d = \text{argmax p}\left(word_i^d \middle| word_{i+3}^u, word_{i+2}^u, word_{i+1}^u, word_i^u\right)$$

Where $word_i^d$ represent the selected diacritized form of the i^{th} un-diacritized word represented by $word_i^u$, given the words next to $word_i^u$ represented by $word_{i+1}^u$, $word_{i+2}^u$ and $word_{i+3}^u$. In case the word was not found in any of the 4-gram models, the system backs off to tri-gram models.

(b) Tri-gram Language Model: Similar to what have been done in the 4-gram model, contexts will continue to be used for diacritizing a certain word. This model has also been split into two sub models, as follows:

(b.1) Right-Context Trigram Model: In this sub-model, the diacritizer will choose $word_i^d$ as the diacritized form of the input word represented by $word_i^u$ considering the previous history of that word represented by $word_{i-1}^u$ and $word_{i-2}^u$, as per the following equation:

$$word_i^d = \text{argmax p}\left(word_i^d \middle| word_i^u, word_{i-1}^u, word_{i-2}^u\right)$$

In case the word was not found to have a right-context tri-gram, the system will back off to a left-context model.

(b.2) Left-Context Trigram: In this case, the diacritizer will choose $word_i^d$ as the diacritized form of the input word represented by $word_i^u$ and the words next to the given one to be diacritized represented by $word_{i+1}^u$ and $word_{i+2}^u$, as per the following equation:

$$word_i^d = \text{argmax p}\left(word_i^d \middle| word_{i+2}^u, word_{i+1}^u, word_i^u\right)$$

In case of the word was not found in any of the tri-gram models, the system backs off to bi-gram models.

(c) Bigram Language Models: Similar to the previous two models, contexts will continue to be used for diacritizing a certain word. This model has also been split into two sub models, as follows:

(c.1) Right-Context Bigram Model: diacritizer will choose $word_i^d$ as per the following equation:

$$word_i^d = \text{argmax p}\left(word_i^d \middle| word_i^u, word_{i-1}^u\right)$$

In case the word was not found to have a right-context bi-gram, system set will back off to a left-context model.

(c.2) Left-Context Bigram Model: diacritizer will choose $word_i^d$ as per the following equation:

$$word_i^d = \text{argmax p}\left(word_i^d \middle| word_{i+1}^u, word_i^u\right)$$

In case of the word was not found in any of the bi-gram models, the system backs off to a unigram model.

(d) Unigram Model: In this case, each un-diacritized word in the test set will be replaced by the corresponding diacritized one that occurs most frequently in the training set, as per the following equation:

$$word_i^d = \text{argmax p}\left(word_i^d \middle| word_i^u\right)$$

By adopting all word level language models as described in the above sections using the dialectal Arabic corpus, the system has resulted in a Diacritization WER and DER of 24.8% and 21.7% as shown in Figure 1.

By adopting all word level language models using MSA corpora, the system has resulted in a Diacritization WER and DER of 39.1% and 28.9% as shown in Figure 2.

By adopting all word level language models using the dialectal Arabic corpus first, then MSA corpora for out-of-vocabulary (OOV) words, the system has resulted in a Diacritization WER and DER of 22.7% and 19.3% as shown in Figure 3.

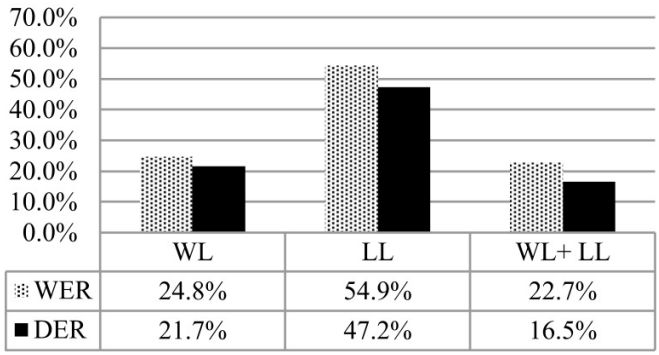
For OOV words, the system backs off to letter-based n-gram modeling.

Letter-Level Lexical Modeling: In this lexical level, the system splits each word into set of letters. Based on that, in this type, and similar to what was done in the previous type - morphemes level - the diacritizer considers each letter in the word as a single unit. For example, the word "العامية" is decomposed into ا – ل – ع – ا – م – ي – ة, hence, four different letter-based models and sub-models that are similar to the word-based and morphological-based models shall be used to re-introduce the missing diacritic marks for each letter for OOV words.

By only applying the letter-based approach using the dialectal Arabic corpus, the system resulted in WER and DER of 54.9% and 47.2% respectively. However, as a result for combining the multi-lexical models: word, morpheme and letter levels, the system resulted in WER and DER of 22.7% and 16.5% respectively as shown in Figure 1.

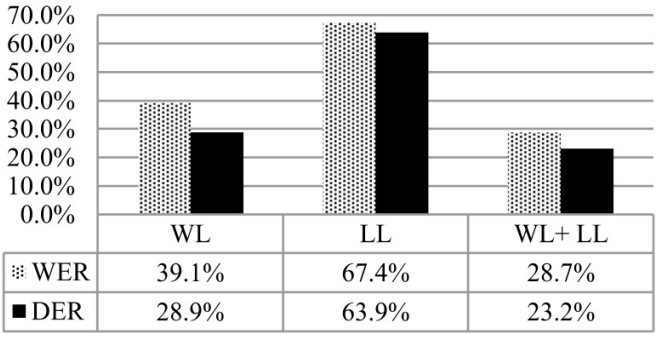
By only applying the letter-based approach using MSA corpora, the system resulted in WER and DER of 67.4% and 63.9% respectively. However, as a result for combining the multi-lexical models: word, morpheme and letter levels, the system resulted in WER and DER of 28.7% and 23.2% respectively as shown in Figure 2.

By only applying the letter-based approach using the dialectal Arabic corpus first, then MSA corpora for out-of-vocabulary (OOV) words, the system resulted in WER and DER of 50.1% and 46.3% respectively. However, as a result for combining the multi-lexical models: word, morpheme and letter levels, the system resulted in WER and DER of 16.8% and 11.7% respectively as shown in Figure 3.



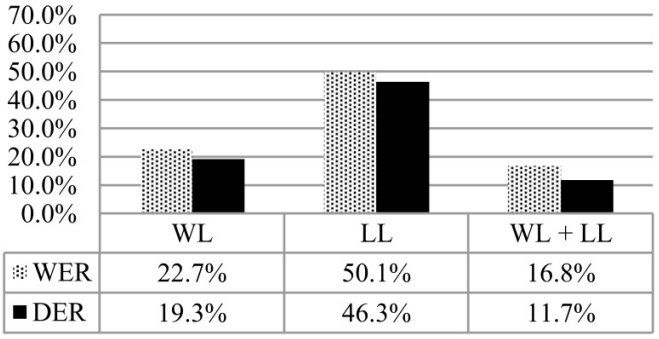
	WL	LL	WL+ LL
WER	24.8%	54.9%	22.7%
DER	21.7%	47.2%	16.5%

Figure 1: WER and DER results for the different applied techniques using the dialectal Arabic corpus, **WL**=Word Level, **LL**= Letter Level.



	WL	LL	WL+ LL
WER	39.1%	67.4%	28.7%
DER	28.9%	63.9%	23.2%

Figure 2: WER and DER results for the different applied techniques using MSA corpora, **WL**=Word Level, **LL**= Letter Level.



	WL	LL	WL + LL
WER	22.7%	50.1%	16.8%
DER	19.3%	46.3%	11.7%

Figure 3: WER and DER results for the different applied techniques using the dialectal Arabic corpus first, then MSA corpora for out-of-vocabulary (OOV) words, **WL**=Word Level, **LL**= Letter Level.

Figure 4 shows a block diagram for the proposed multi-lexical level automatic Diacritization system for Arabic.

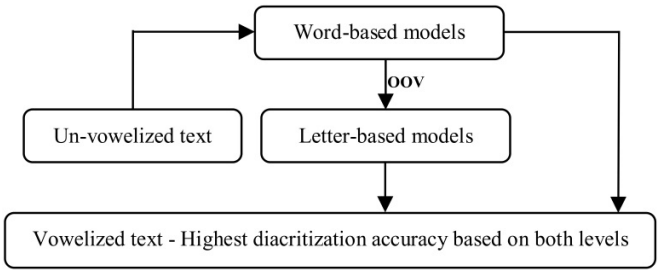


Figure 4: A block diagram for the proposed automatic diacritization system.

5. Conclusions

This paper has presented a multi-lexical level statistical approach for Automatic diacritization for dialectal Arabic. Our system is based statistical n-grams approach, in order to achieve better diacritization accuracy from dialectal Arabic text.

Several configurations were also investigated. The best configuration was the diacritization through dialectal Arabic word-level, MSA word-level, then dialectal Arabic letter-level, with consideration of sub-models for each one. The best reported results were a WER of 16.8% and DER of 11.7%.

For future work, we are plaining to expand our training set data to cover other dialectal Arabic forms.

References

[1] M. Elmahdy, R. Gruhn and W. Minker, "Novel Techniques for Dialectal Arabic Speech Recognition," ISBN 978-1-4614-1906-8, Springer New York Dordrecht Heidelberg London, 2012.

[2] M. Elmahdy, R. Gruhn, S. Abdennadher and W. Minker, "Rapid Phonetic Transcription using Everyday Life Natural Chat Alphabet Orthography for Dialectal Arabic Speech Recognition," The IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp. 4936-4939, Prague, 2011.

[3] N. Habash and O. Rambow, "Arabic Diacritization through Full Morphological Tagging," in Proceedings of NAACL HLT, Companion Volume, pp. 53-56, 2007.

[4] A. Al Sallab, M. Rashwan, H. M. Raafat, and A. Rafea, "Automatic Arabic Diacritics Restoration Based on Deep Nets," In Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP), pp. 65-72, 2014.

[5] I. Zitouni and R. Sarikaya, "Arabic Diacritic Restoration Approach Based on Maximum Entropy Models," in Journal of Computer Speech and Language, Elsevier, 23, pp. 257-276, 2009.

[6] M. Atiyya, K. Choukri, and M. Yaseen, "NEMLAR Arabic Written Corpus," The NEMLAR project, 2005.medar.info/The_Nemlar_Project/Publications/WC_design_final.pdf

[7] M. Rashwan, M. Al-Badrashiny, M. Attia, S. M. Abdou, and A. Rafea: "A Stochastic Arabic Diacritizer Based on a Hybrid of Factorized and Unfactorized Textual Features," in IEEE Transactions on Audio, Speech and Language Processing, 19(1), pp. 166-175, 2011.

[8] M. Alghamdi and Z. Muzaffar, "KACST Arabic Diacritizer" In The First International Symposium on Computers and Arabic Language, pp. 25-28, 2007.

[9] Shaalan, K., Abo Bakr, H., & Ziedan, I. (2007). Transferring Egyptian Colloquial into Modern Standard Arabic, International Conference on Recent Advances in Natural Language Processing, pp. 525-529, Bulgaria.

[10] A. Al-Taani and S. Abu Al-Rub, "A Rule-Based Approach for Tagging Non-Vocalized Arabic Words," The International Arab Journal of Information Technology, vol. Vol. 6, 2009.

[11] N. Habash and O. Rambow, "Arabic Diacritization through Full Morphological Tagging," in Proceedings of NAACL HLT 2007, Companion Volume, pp. 53-56, Rochester, NY, 2007.

[12] M. Ameur, Y. Moulahoum and A. Guessoum, "Restoration of Arabic Diacritics Using a Multilevel Statistical Model," in IFIP International Federation for Information Processing, 2015.

[13] S. Harrat , M. Abbas , K. Meftouh and K. Smaïli, "Diacritics Restoration for Arabic Dialects," in 14th Annual Conference of the International Speech Communication Association , Lyon, France, 2013.

[14] Y. Hifny, "Restoration of Arabic Diacritics using Dynamic Programming," IEEE, pp. 978-1-4799-0080-0/13, 2013.

[15] Y. Hifny, "Higher Order n-gram Language Models for Arabic Diacritics Restoration," in The Twelfth Conference on Language Engineering, Cairo, Egypt, 2012.

[16] Y. Hifny, "Smoothing Techniques for Arabic Diacritics Restoration," in The Twelfth Conference on Language Engineering, Cairo, Egypt, 2012.

[17] M. Rashwan, A. Al Sallab, H. Raafat and A. Rafea, "Deep Learning Framework with Confused Sub-Set Resolution Architecture for Automatic Arabic Diacritization," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. VOL. 23, 2015.

[18] G. Abandah, A. Graves and B. Al-Shag, "Automatic diacritization of Arabic text using recurrent neural networks," International Journal on Document Analysis and Recognition (IJDAR), vol. 18, no. Issue 2, pp. pp 183-197, 2015.

[19] LDC Arabic Tree Bank Part 3, [Online]. Available: http://www.ldc.upenn.edu/Catalog/catalogEntry.jsp?catalogId=LDC2005T20

[20] H. Gadalla, H. Kilany, H. Arram, A. Yacoub, A. El-Habashi, A. Shalaby, K. Karins, E. Rowson, R. MacIntyre, P. Kingsbury, D. Graff, C. McLemore, "CALLHOME Egyptian Arabic Transcripts," Linguistic Data Consortium (LDC), LDC catalog no. LDC97T19, 1997.

[21] A. Zayyan, M. Elmahdy, H. binti Husni, J. Al-Ja'am, "Automatic Diacritics Restoration for Modern Standard Arabic Text," IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE), Penang, Malaysia 2016.

Ayman A. Zayyan is currently Teaching Assistant at Qatar University, Department of Computer Science and Engineering. He is working towards his Master degree at College of Arts and Science - University Utara Malaysia. His research is focused on Automatic diacritization for un-diacritized Arabic text.

Dr.-Ing. Mohamed Elmahdy is currently an assistant professor at the German University in Cairo. He received his Ph.D. from Ulm University, Germany, in 2011. He received his B.Sc. and M.Sc. years 2000 and 2004 respectively in Engineering from Cairo University. From 2000 to 2006, he has been working as an R&D engineer at IBE Technologies, Cairo, Egypt. From 2007 to 2011, he was pursuing his Ph.D. degree at the Dialogue Systems Group, Institute of Information Technology at the University of Ulm in cooperation with the German University in Cairo. From 2011 to 2014, he worked as a postdoctoral research fellow at Qatar University in cooperation with the University of Illinois at Urbana-Champaign. His research interest is mainly focused on natural language processing and machine learning.

Dr. Husniza Husni received her PhD degree from Universiti Utara Malaysia in 2010. She is currently a senior lecturer at School of Computing, Universiti Utara Malaysia and a member of Human-Centered Computing research lab. Her research interests include speech recognition for dyslexic children reading, interaction design for children, child-robot interaction, user experience, educational technology, artificial intelligence, and game based learning. She has won several innovative awards for her educational technology products. She taught several courses on artificial intelligence techniques, including fuzzy logic, introduction to artificial intelligence, expert system, intelligent agent systems. She now teaches research methodology in IT and coordinates projects for final year students.

Prof. Jihad Mohamed Alja'am received the Ph.D. degree, MS. degree and BSc degree in computing from Southern University (The National Council for Scientific Research, CNRS), France. He was with IBM-Paris as Project Manager and with RTS-France as IT Consultant for several years. He is currently with the Department of Computer Science and Engineering at Qatar University. His current research interests include multimedia, assistive technology, learning systems, human–computer interaction, stochastic algorithms, artificial intelligence, information retrieval, and natural language processing. Dr. Alja'am is a member of the editorial boards of the Journal of Soft Computing, American Journal of Applied Sciences, Journal of Computing and Information Sciences, Journal of Computing and Information Technology, and Journal of Emerging Technologies in Web Intelligence. He acted as a scientific committee member of different international conferences (ACIT, SETIT, ICTTA, ACTEA, ICLAN, ICCCE, MESM, ICENCO, GMAG, CGIV, ICICS, and ICOST). He is a regular reviewer for the ACM computing review and the journal of supercomputing. He has collaborated with different researchers in Canada, France, Malaysia, and USA. He published so far 138 papers, 8 books chapters in computing and information technology which are published in conference proceedings, scientific books, and international journals.