# Diacritics restoration for Arabic dialect texts

*S. Harrat*[1], *M. Abbas*[2], *K. Meftouh*[3], *K. Smaili*[4]

[1]ENS Bouzareah, Algiers, Algeria
[2]CRSTDLA, Algiers, Algeria
[3]Badji Mokhtar University-Annaba, Algeria
[4]Campus scientifique LORIA , Nancy, France

slmhrrt@gmail.com, m_abbas04@yahoo.fr, Karima.meftouh@univ-annaba.org, smaili@loria.fr

## Abstract

In this paper we present a statistical approach for automatic diacritization of Algiers dialectal texts. This approach is based on statistical machine translation. We first investigate this approach on Modern Standard Arabic (MSA) texts using several data sources and extrapolated the results on available dialectal texts. For evaluation we used word and diacritization error rates and also precision and recall.

**Index Terms**: Machine translation system, Modern Standard Arabic, automatic diacritization, Algiers's dialect.

## 1. Introduction

Vocalization, diactritization or diacritics restoration is one of the major challenges for the Arabic natural language processing. Indeed, the absence of vowels in Arabic texts generates a considerable number of ambiguities in morphological, syntactic and semantic layer. For Arabic speakers, absence of vowels does not pose problems for understanding texts, although in some cases vocalization is not intuitive. Automatic vocalization for Arabic language (even for other languages) remains an unresolved problem for which several works are dedicated [1][2][3][4]. Absence of diacritics in Arabic texts produces a serious problem for many applications such as grapheme-phoneme conversion. Algiers dialect as an Arabic language is concerned by this problem which produces several challenges for many automatic tasks on this language. The goal of this work is automatic diacritics restoration for Algiers dialect texts. Actually, this is an intermediate purpose, since our general goal is speech translation between Modern Standard Arabic and Algerian dialect. In order to reach this aim, translation system must include a Text-to-Speech module. The first component of this module is a grapheme-phoneme converter that we have to develop. Results of grapheme phoneme-conversion for Arabic texts (both classical and dialectal) could not be acceptable if these texts are not diacritized. That is why the first step for developing our grapheme-phoneme converter is an automatic diacritics restoration for Algiers dialect texts.

In this paper, we present an automatic diacritization system for Arabic texts based on statistical approach. We attempt to use available tools for statistical machine translation for building such a system which basically does not require any linguistic knowledge. We began by working on MSA texts for many reasons: Algiers dialect is an Arabic language which obeys to almost the same writing rules of MSA. Availability of diacritized texts in MSA allows to test our system on a large amount of data, which is not the case for Algiers dialect. Finally, we worked first on MSA texts because of the available results for many works in this field.

This paper is organised as follows: In section 2, we describe the Arabic language and present the main features of this language especially those related to diacritization. In section 3, we provide some specificities of Algiers dialect. In section 4, we present related works in this field summarizing the main aspects of every work. In section 5, we present our approach and describe the used data (for training, tuning and testing). Evaluation results and experiments are reported in the next. Finally, We conclude by showing our main results and future work directions.

## 2. Arabic language

Arabic is a Semitic Language with consonantal alphabet (which denotes only consonants). Arabic alphabet consists of 28 letters [1] denoting consonants and three long vowels (ا *ā*, و *w* and ي *y*). Arabic script includes short vowels ( ـَ *a*, ـُ *u*, ـِ *i* ) and other phonetic symbols which are represented by diacritics (strokes placed above or below consonants and long vowels). Short vowels can appear anywhere in the word, the *tanween* represents doubled case endings, it consists of three cases (ـً *an*, ـٌ *un*, ـٍ *in* ) it appears only in the end of the words and expresses nominal indefiniteness. Arabic diacritics include also syllabification marks: the *Shadda* or gemination mark denotes a double consonant(it could be combined with short vowels and doubled case endings), and the *Sukun* which denotes the absence of vowel (see Table 1). According to the function of the word in the sentence, an Arabic word could take as case endings the short vowels ( ـَ *a*, ـُ *u*, ـِ *i* ) respectively for accusative, nominative and genitive case. It is important to note that in some cases diacritics are not allowed such as in the case of:

- The letter ى *ā (Alif maqsura)* which never takes diacritics such as in the word سَعَى *saʿā* (to endeavor).
- The long vowels و *w* and ي *y* when preceded (in a word) respectively by the short vowels ـُ *u* and ـِ *i*.
- The long vowel ا *ā* when preceded (in a word) by ـً *an* and in the definite article الـ *āl*.
- The consonant لـ *l* does not take a diacritic in the definite article الـ *āl* when it is followed by a solar consonant.

[1]Including 14 solar consonants which assimilate the ل of a preceding definite article أل and 14 lunar consonants which do not assimilate it.

Table 1: Arabic diacritics and their pronunciation

| Diacritized consonant /b/ | Name | Pronunciation |
|---|---|---|
| بَ | *Fatha* | /ba/ |
| بُ | *Damma* | /bu/ |
| بِ | *Kasra* | /bi/ |
| بًا | *Tanween Fatha* | /ban/ |
| بٌ | *Tanween damma* | /bun/ |
| بٍ | *Tanween kasra* | /bin/ |
| بّ | *Shadda* | /bb/ |
| بْ | *Sukun* | /b/ |

Arabic diacritics are used for disambiguation. A word without diacritics could have many valid diactritizations (depending on its grammatical category) which generates several interpretations (see Table 2 for some examples).

Table 2: Some possible diacritizations of Arabic word فسر *fsr*

| Diacritic form | Meaning | Grammatical category |
|---|---|---|
| فَسَّرَ *fassara* | he explained | verb (active voice) |
| فُسِّرَ *fussira* | was explained | verb (passive voice) |
| فَسِرْ *fasir* | so walk | conjunction+ verb |
| فَسِرٌّ *fasirrun* | and a secret | conjunction+noun |
| فَسْرٌ *fasrun* | statement | Noun |

## 3. Algiers dialect

In Algeria, as elsewhere, spoken Arabic differs from written Arabic; Algerian Arabic has a vocabulary inspired from Arabic but the original words have been altered phonologically[5]. The Algiers dialect represents the dialectal Arabic spoken in Algiers and its periphery. This dialect is different from the dialects spoken in the other areas of the country. Algiers dialect simplifies the morphological and syntactic rules of the written Arabic. It uses the Arabic alphabet which includes 28 letters; however the use of some letters like ظ *ẓ* and ذ *ḏ* is very rare, most of the time ظ *ẓ* is pronounced ض *ḍ* and ذ *ḏ* is pronounced د *d*. The same case is observed for the letter ث *ṯ* which is pronounced ت *t*. Moreover Algiers dialect uses some non-Arabic letters like پ *p* as in the word پاپا *pāpā* (dad). Besides phonological alteration of words, Algiers dialect drops the case endings of the written language. Also, it uses Arabic script which denotes consonants and diacritics. It uses all Arabic diacritics listed above except the *Tanween* doubled case endings. We notice that case endings of words in Algiers dialect are replaced by the *Sukun* (absence of vowel), this simplifies diacritization process but generates ambiguity at syntactic level. This conducts to an ambiguity concerning the identification of grammatical role of certain words, for example, to specify the agent and the object in some simple sentences without using semantic knowledge (see in Table 3 a detailed example). Like written Arabic, diacritics in Algiers dialect are used for disambiguation, a word without diacritics could accept many valid diacritizations and then if they are missing, it would be difficult to understand its meaning(see Table 4 for some examples). It is important to note that, due mainly to dropping case endings of words in Algiers dialect, the number of multiple valid diacritizations for a word is less important than it is for an MSA word.

## 4. Related works

Most works on diacritization are dedicated for Modern standard Arabic. Several approaches were adopted, in [1] the system is based on hierarchical search at sentence, phrase, word and character level. From the sentence level, the system attempts to retrieve diacritized examples from training data, if no suitable example is found for the given sentence, the system splits it into phrases and searches for fitting diacritics phrases in the training data. If the search fails the phrase is split into words then into characters (if word search fails). The system uses n-grams to define suitable examples. In [2] the diacritization was considered as a machine translating problem from non-diacritized texts to diacritized texts. The authors adopted a rule based approach. The shortcomings of this system are those of rule based translation systems: adding new cases requires the definition of new rules which makes system maintenance expensive in terms of human efforts. In [3] the authors combined weighted finite state machines and language models. Their system is based on three language models (trained on LDC's Arabic Treebank Data): word, character and clitics. Transducers transitions are obtained by probabilistic calculations from the three language models. In [4] two approaches are used, the first one a SMT based approach combined with rule based diacritizer, the second considers diacritization as a sequence labelling problem. The first approach operates at the whole sentence, word level and then at the character level then a combination of the two (word and character) levels is achieved. A statistical phrase based system is built by the alignment between the output of rule based system and correct diacritized text in order to perform a post-editing step. The second approach based on sequence labelling problem used Conditional Random Fields (CRFs) in order to predict the correct sequence of diacritics for un-diacritized consonants sequence. In [6] and [7] Markov Models (HMM) are used for diacritization, non-diacritized words are considered as observation and their possible diacritizations are taken as the hidden states that produced these observations. The Viterbi algorithm is used to define the best hidden states. It should be noted that these two studies use the Koran for training and testing. In [8] the diacritization is viewed as sequence classification problem: given a sequence of characters X, each character is labelled by its diacritic (a sequence of labels Y is obtained). The objective of the system is to assign the sequence of labels Y to X sequence of consonants, the authors propose a statistical approach based on maximum entropy framework (MaxEnt henceforth). The classification features used by their system were lexical, segment-based and part-of-speech tags.

## 5. Baseline system

Our approach is based on statistical machine translation where source language is unvocalized text and target language is vocalized text.

Table 3: Example of ambiguity generated by dropped case endings in Algiers dialect

| Dialectal Sentence | Arabic Sentence | Meaning | Agent | Object |
|---|---|---|---|---|
| سْمَعْ وَلِيدْ عُمَرْ *smaʿwalīd ʿumar* | سَمِعَ وَلِيدٌ عُمَرَ *samiʿa walīdun ʿumara* | Walid heard Omar | Walid | Omar |
| | سَمِعَ وَلِيدًا عُمَرُ *samiʿa walīdan ʿumaru* | Omar heard Walid | Omar | Walid |

Table 4: Example of multiple diacritizations of dialectical words

| Dialectal Word | Valid diacritization | Meaning |
|---|---|---|
| جوز *ğūz* | جَوَّزْ *ğawwaz* | You or he spent |
| | جُوزْ *ğuwz* | You pass |
| | جُوزْ *ğuwz* | Nut |
| قريت *qrīt* | قْرِيتْ *qrīt* | I or you read/studied |
| | قَرِّيتْ *qarrīt* | I or you teached |

### 5.1. Building the SMT system for diacritization

We built a SMT system for diacritization based on parallel corpora of diacritized and undiacritized texts. The system is phrase based [9] with default settings: bidirectional phrase and lexical translation probabilities, distortion model (with seven features), a word and a phrase penalty and a language model. The system consists of a word alignment between source and target which is got by GIZA++ [11], a phrase table with undiacritized and diacritized entries, a trigram language model trained on diacritized texts, we use for this purpose SRILM toolkit[10].

### 5.2. The Data

Using SMT approach assumes the availability of the parallel corpora for source and target languages. To build such a resource in our case, it is enough to get a vocalized corpus when available and proceed to remove diacritics from it or in the other case, diacritize an unvolcalized corpus. The two ways were exploited in our work, the first one for MSA and the second one for Algiers dialect.

#### *5.2.1. MSA corpora*

At the beginning of this work, the only available corpus for us was Tashkeela[2] a free corpus under GPL licence. This corpus is a collection of classical Arabic books downloaded from an on-line library [3]. It consists of more than 6M words. We began by removing special symbols, numbers and some non Arabic characters from vocalized corpus, then we split long sentences to shorter ones. After that we removed the diacritics from it to get the unvocalised corpus. The training data consists of about 1200K pairs of sentences. It is important to note that we split data on training (80%), developing(10%) and testing sets(10%), by randomly allocating a set of books to each task. For comparison purpose, We get the LDC Arabic Treebank (Part3, V1.0)[12], which is widely used in several NLP systems especially those dedicated for diacritics restoration[3][4][8]. This corpus is a set of 600 documents collected from Annahar News Texts. It includes 340K Words. To exploit this resource, we built a full vocalized corpus by exploring all documents part-of-speech tagging, we extracted for each word in a document its correspondent diacritization (manually annotated by LDC team). For splitting the corpus into training developing and testing sets we followed the same repartition as tashkeela by randomly allocating the documents to each set.

#### *5.2.2. Algiers dialect corpus*

The big challenge of this work is the availability of Algiers dialect corpus since this language is a non-resourced language. We began by creating this corpus by hand, initially it did not contain diacritics, so we proceed to vocalize it by hand also. This task was expensive in terms of time and human effort. At this time, the vocalized corpus consists of 4K pairs of sentences, with 23K words.

## 6. Experiments

For the evaluation of the system, we use WER (Word Error Rate): the percentage of incorrectly diacritization word (delimited by white-space). At the character level, we use DER (Diacritization Error Rate). We compute these error rates by using Sclite[4] part of NIST SCTK Scoring Toolkit which finds alignments between reference and hypothesis for both word and character levels. A word is considered as incorrectly diacritized if at least one of its characters have an incorrect diacritization. Since WER and DER are relative to error rates, we were interested by evaluating the results in terms of correctness rate. For this purpose we computed precision and recall. These measures are largely used for evaluating many Information Retrieval Systems. For computing these values, we consider that a set of candidate items are compared to a set of reference item, in the case of evaluating our SMT based diacritizer, these measures could be calculated as follows:

$$Precision = \frac{\#Word\ correctly\ diacritized}{\#Word\ Diacritized} \quad (1)$$

$$Recall = \frac{\#Word\ correctly\ diacritized}{\#Total\ of\ Words} \quad (2)$$

We notice that recall rate could be deduced from WER, it corresponds to:

$$Recall_{word} = 100 - WER \quad (3)$$

[2]http://sourceforge.net/project/tashkeela

[3]http://www.shamela.ws

[4]http://www1.icsi.berkeley.edu/Speech/docs/sctk-1.2/sclite.htm

At character level, precision and recall are calculated as above by replacing word count by character count. Recall rate at character level corresponds also to:

$$Recall_{char} = 100 - DER \quad (4)$$

### 6.1. Results for MSA

Results in Table 5 show a WER of 16.2 and a DER of 4.1 for Tashkeela corpus and a WER of 23.1 and a DER of 5.7 for LDC Arabic Treebank corpus. Deletion and insertion error rates are both equal to zero because the same number of words is contained both in test and reference data. The difference between the two sets is only in diacritics. That is why, in DER distribution we notice different deletion and insertion rates. When observing the test data, we notice that the most errors are concentrated on case endings of words relative to short vowels and doubled case endings. Actually, we ran a test ignoring them, the WER and DER decrease with more than 50% for both corpora.

Table 5: Word and Diacritization Error Rate Summary(MSA)

| Corpus | Taskeela | | LDC ATB | |
|---|---|---|---|---|
| Distribution | WER | DER | WER | DER |
| Substitution | 16.2 | 1.8 | 23.1 | 0.5 |
| Deletion | 0.0 | 1.9 | 0.0 | 5.1 |
| Insertion | 0.0 | 0.3 | 0.0 | 0.1 |
| Total(WER/DER) | 16.2 | 4.1 | 23.1 | 5.7 |

In terms of precision (we remind that recall is deducted from WER), we notice slight decrease of Tashkeela rates compared with ATB rates (see Table 6).

Table 6: Precision and recall Summary(MSA)

| Level | Word | | Character | |
|---|---|---|---|---|
| Corpus | Recall | Precision | Recall | Precision |
| Taskeela | 83.8% | 85.2% | 95.9% | 93.1% |
| ATB | 76.9% | 89.2% | 94.3% | 96% |

### 6.2. Results for Algiers dialect

For Algiers dialect experiments, we notice a higher WER and DER than for MSA experiments. But this is mainly due to the smallness of the Algiers dialect corpus. We get a WER of 25.8 and a DER of 12.8 (see in Table 7 detailed results). It is important to note that the DER at character level is mostly concentrated on the deletion rate, we notice also a small proportion of substitution. The high deletion rate is due to absence of words in the training data.

Table 7: Word and Diacritization Error Rate Summary (Algiers dialect)

| Distribution | WER | DER |
|---|---|---|
| Substitution | 25.8 | 0.1 |
| Deletion | 0.0 | 12.6 |
| Insertion | 0.0 | 0.1 |
| Total(WER/DER) | 25.8 | 12.8 |

We computed also recall and precision values from test data. At word level, the precision and recall where respectively 96.3% and 74.2,%. We notice a high precision compared to recorded rates for MSA corpora, this is due mainly to absence of words case endings in Algiers dialect and the multiple diacritization of a word in this language is less important than it is in MSA. At character level, values recorded were 98% for precision and 87.2% for recall (see Table 8 ).

Table 8: Precision and recall Summary(Algiers dialect)

| Level | Word | | Character | |
|---|---|---|---|---|
| Corpus | Recall | Precision | Recall | Precision |
| Algiers dialect | 74.2% | 96.3% | 87.2% | 98% |

We have performed several tests by increasing the amount of data at every test, and we observed that when the corpus size increases by a little percentage, the WER and DER decrease also. For comparison, we also run many tests for small MSA corpora extracted from Tashkeela and ATB with the same size order as Algiers dialect corpus, recorded WERs and DERs were respectively more than 56.5 and 20.5 at each time and for both corpus.

## 7. Conclusion

We presented a statistical approach for diacritization of Algiers dialect texts based on a machine translation system. We experimented the solution on MSA corpora, for positioning our results against other works. We did not use any other NLP tools (even if they are available for Arabic language) in order to be in the same conditions for Algiers dialect which is an under-resourced language. Regards to the small amount of training data for the dialect, we got an acceptable WER and DER when we compared them to the results with small MSA corpora. For Arabic language side, the results for Tashkeela corpus were slightly better compared to those for Arabic Treebank corpus. This is mainly due to the size of the two corpora and the nature of data of each one. In fact, the ATB consists of several kinds of texts with entirely different topics, data disparity in its case is considerable, whereas Tashkeela corpus contains classical books related to theology. In terms of precision, we notice that inspite of the small amount of available data for the dialect, we got a higher percentage compared to MSA corpora (96.3% vs. 85.2% and 89.2%) at word level and (98% vs. 93.1% and 96%) at character level. Considering this interesting precision rate, we will use this system for the vocalization of the rest of the dialectal corpus, this will save human effort and time since a little amount of the data needs to be corrected (by hand). We will use this iterative process for enriching our dialect corpus for getting better results with few efforts.

# 8. References


[1] O. Emam and V. Fischer, "Hierarchical Approach for the Statistical Vowelization of Arabic Text. Technical report", in IBM Corporation Intellectual Property Law, Austin, TX, US, 2005.

[2] T.A El-Sadany and M.A Hashish, "An Arabic Morphological System", IBM Systems Journal, 28(4), 1989.

[3] R. Nelken and S.M. Sheiber, "Arabic Diacritization Using Weighted Finite-State Transducers.", Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages, Ann Arbor, Michigan, Ann Arbor, Michigan, USA, 2005.

[4] T. Schlippe, T. Nguyen and S. Vogel, "Diacritization as a Machine Translating 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.

[5] K. Meftouh, N. Bouchemal and K. Smaïli, "A study of a non-resourced language: an Algerian dialect". In proceedings of the third international workshop on spoken languages technologies for under-resourced languages. Cape town, South Africa, 2012.

[6] Y. Gal, "An HMM approach to vowel restoration in Arabic and Hebrew", Proceedings of the ACL-02 workshop on Computational approaches to Semitic languages, PA, USA, 2002.

[7] H. Al Muhtaseb, M. Elshafei and M. Al-Ghamidi, "Statistical Methods for Automatic Diacritization of Arabic Text", Proceedings of the Saudi 18th National Computer Conference (NCC18), Riyadh, Saudi Arabia, 2006.

[8] I. Zitouni, J.S Sorensen and R. Sarikaya, "Maximum entropy based restoration of arabic diacritics, Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL, Sydney, Australia, 2006.

[9] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin and E. Herbst. "Moses: Open Source Toolkit for Statistical Machine Translation", Annual Meeting of the Association for Computational Linguistics (ACL), demonstration session, Prague, Czech Republic, 2007.

[10] A. Stolcke, "SRILM An Extensible Language Modeling Toolkit on Spoken Language Processing", Proceedings of the International Conference, volume 2, pp. 901-904, Denver, 2002.

[11] F. Josef Och, H. Ney, "A Systematic Comparison of Various Statistical Alignment Models, Computational Linguistics, volume 29, number 1, pp. 19-51, 2003.

[12] M. Maamouri, A. Bies, T. Buckwalter and H. Jin , "Arabic Treebank: Part 3 v 1.0", Linguistic Data Consortium, Philadelphia, 2004.