Exploiting Arabic Diacritization for High Quality Automatic Annotation

Nizar Habash, Anas Shahrour, Muhamed Al-Khalil

Exploiting Arabic Diacritization for High Quality Automatic Annotation

Nizar Habash, Anas Shahrour, Muhamed Al-Khalil
Computational Approaches to Modeling Language Lab
New York University Abu Dhabi, UAE
{nizar.habash,anas.shahrour,muhamed.alkhalil}@nyu.edu

Abstract

We present a novel technique for Arabic morphological annotation. The technique utilizes diacritization to produce morphological annotations of quality comparable to human annotators. Although Arabic text is generally written without diacritics, diacritization is already available for large corpora of Arabic text in several genres. Furthermore, diacritization can be generated at a low cost for new text as it does not require specialized training beyond what educated Arabic typists know. The basic approach is to enrich the input to a state-of-the-art Arabic morphological analyzer with word diacritics (full or partial) to enhance its performance. When applied to fully diacritized text, our approach produces annotations with an accuracy of over 97% on lemma, part-of-speech, and tokenization combined.

Keywords: Arabic, Morphology, Annotation, Diacritization

1. Introduction

The automatic processing of the Arabic language is challenging for a number of reasons. Two inter-related reasons are Arabic's complex morphology and its diacritic-optional orthography (Habash, 2010). This combination leads to the high number of about 12.8 possible analyses per word out of context (Shahrour et al., 2015). Much research has been done on Arabic morphological disambiguation and automatic diacritization. Most of this work relies on existing morphologically annotated (i.e., contextually disambiguated) corpora. Some researchers interested only in diacritization also use diacritized corpora. The morphologically annotated corpora for Arabic are limited by size and genre. This is because the cost of creating such corpora is prohibitive as time and money must be dedicated to collect data, create guidelines, hire and train annotators, and continuously evaluate the quality of the annotations.

In this paper we explore an alternative approach to Arabic morphological annotation: we exploit fully diacritized text to rerank the morphological analyses of a state-of-the-art morphological tagger that expects no diacritics as input and provides not only part-of-speech, morphological features and lemmas, but also diacritizations. We evaluate and analyze the accuracy of this method in generating high quality morphologically analyzed corpora that can themselves be used for improving tools for Arabic processing, especially for less studied genres. The approach relying solely on diacritization is also very cheap in comparison to hiring specialized (linguist) annotators and training them. Although educated Arabic speakers do not use diacritics in their writing they know how to assign them reasonably well. Of course, we still expect the diacritic annotators to be well educated in Standard Arabic grammar and be fast typists. But the supply and demand, as well as the limited overhead of training is in our favor.[1] Obviously, this kind of approach can work only because we already have a mass of Arabic resources (the taggers and analyzers) that facilitate it. The idea is also applicable to other languages with similar orthographic ambiguity challenges, e.g., Hebrew and Persian. Our results on Arabic show that using our approach produces quality comparable to human manual annotation (Maamouri et al., 2008; Habash and Roth, 2009) and other automatic enhancement techniques (Alkuhlani et al., 2013), reaching an accuracy of over 97% on lemma, part-of-speech, and tokenization combined.

We present next some background Arabic linguistic facts and related work, followed by a discussion of our approach and a detailed evaluation of it on different genres and different degrees of diacritization.

2. Arabic Linguistic Facts

The Arabic script consists of two classes of symbols: letters and diacritics (Habash and Rambow, 2007; Habash, 2010). Whereas letters are always written, diacritics are optional. Written Arabic text can be undiacritized, partially, or fully diacritized. Religious text such as the Holy Qur'an are fully diacritized. However, in newswire text, 1.6% of all words have at least one diacritic indicated by their author, mostly to disambiguate the text for readers (Habash, 2010).

There are three types of diacritics: vowel, nunation, and shadda. Vowel diacritics represent Arabic's three short vowels (/a/, /i/, /u/) and the absence of any vowel. The following is the same word with and without these four diacritics: مبتسم *mbtsm*[2] / مُبْتَسِم *mubtasim* 'smiling'. The three nunation diacritics can only occur in word final positions in nominals (nouns, adjectives and adverbs). They represent a short vowel followed by an /n/ sound, and indicate nominal indefiniteness, e.g., مُبْتَسِمٌ *mubtasim***ũ** /mubtasim**un**/ 'smiling [nominative indefinite]'. The Shadda diacritic is a consonant doubling marker, and can be com-

[1] For reference, a professional Arabic typist can type around 2,250 words/hour undiacritized and 660 words/hour diacritized. S/he can start working immediately to produce diacrtitized text and requires no training or special interfaces.

[2] Arabic transliteration is presented in the Habash-Soudi-Buckwalter scheme (Habash et al., 2007b): (in alphabetical order)
أ ب ت ث ج ح خ د ذ ر ز س ش ص ض ط ظ ع غ ف ق ك ل م ن ه و ي
Â b t θ j H x dð r z s š S D T Ď ς γ f q k l m n h w y
and the additional symbols: ' ء, Â أ, Ă إ, Ā آ, ŵ ؤ, ŷ ئ, ى ,ħ ة, ý ى.

bined with vowel or nunation diacritics, e.g., compare كَتَبَ *kataba* (no Shadda) 'he wrote', with كَتَّبَ *kat∼aba* /kattaba/ (with Shadda) 'he dictated'.

Functionally, diacritics can be split into: *lexemic diacritics* and *inflectional diacritics* (Habash and Rambow, 2007; Habash, 2010). Lexemic diacritics distinguish between two lexemes. For example, the diacritization difference between the lexemes كَاتِب *kAtib* 'writer' and كَاتَب *kAtab* 'he corresponded' distinguish between the meanings of the word rather than their inflections. Inflectional diacritics distinguish different inflected forms (morpho-syntactic variants) of the same lexeme. For instance, the following three forms of the word كاتب *kAtb* 'writer' vary in terms of their inflectional case and state: كَاتِبُ *kAtibu* '[nominative definite]', كَاتِبٌ *kAtibũ* '[nominative indefinite]', كَاتِبِ *kAtibi* '[genitive definite]', and كَاتِبٍ *kAtibĩ* '[genitive indefinite]'. While the lack of diacritics does not seriously hinder educated Arabic speakers, it is a serious cause of ambiguity and a challenge to automatic processing. The state-of-the-art morphological tagger MADAMIRA (Pasha et al., 2014) reports a lexical choice accuracy of about 96% and a full diacritization accuracy of about 86%. Inflectional diacritics, and specifically nominal case diacritics are particularly challenging. Recently, Shahrour et al. (2015) demonstrated that the use of syntactic parsing can help improve the diacritization choice. For more information on Arabic diacritization and morphology, see (Habash and Rambow, 2007; Habash et al., 2007a; Habash, 2010).

3. Related Work

There have been many notable efforts on the development of annotated Arabic language corpora (Maamouri and Cieri, 2002; Maamouri et al., 2004; Smrž and Hajič, 2006; Habash and Roth, 2009; Alansary and Nagi, 2014). However, most contributions adopted manual annotation and morphological features, which is a time-consuming process that requires lots of training for annotators. The effort to build the Penn Arabic Treebank at the Linguistic Data Consortium (Maamouri and Cieri, 2002; Maamouri et al., 2004), made active use of morphological analyzers (Buckwalter, 2004; Graff et al., 2009) to jointly select diacritizations and morphological tags.

Much work has been done on Arabic diacritization (Vergyri and Kirchhoff, 2004; Nelken and Shieber, 2005; Zitouni et al., 2006; Habash and Rambow, 2007; Alghamdi and Muzafar, 2007; Rashwan et al., 2009; Bebah et al., 2014; Hadj Ameur et al., 2015; Abandah et al., 2015; Bouamor et al., 2015; Shahrour et al., 2015; Belinkov and Glass, 2015). While previous approaches focused on improving the quality of automatic diacritization, sometimes through the use of improved morphological and syntactic feature predictions, this work utilizes diacritization to generate high quality predictions of morphological features.

A method for automatic morphological enrichment of a morphologically underspecified treebank has was presented in Alkuhlani et al. (2013), extending previous efforts on case prediction over syntactic trees by (Habash et al., 2007a). Our work relates to their work by the virtue of using limited information from annotators (in our case, *diacritizers*) to automatically generate annotations for all morphological features. Building a treebank however is time-consuming since it requires training annotators for specific tools and conventions, while typing diacritized text can be quickly done by an educated Arabic typist.

4. Approach

Undiacritized Arabic words are highly ambiguous: in our training data, words had an average of 12.8 analyses per word, most of which are associated with different diacritizations. **Morphological analysis** refers to the process of determining all the possible morphological analyses for a specific word out of context. Each Analysis has a single diacritized form, part-of-speech, and other morphological features. Table 1 shows a list of analyses produced by the morphological analyzer for the word بين *byn*. **Morphological tagging** (*aka* morphological disambiguation) refers to the choosing of a morphological analysis in a specific context.

Our approach is to generate high-quality automatic morphological tagging output by exploiting full diacritization information. Our baseline system is the state-of-the-art Arabic morphological tagger MADAMIRA (Pasha et al., 2014) which uses the Standard Arabic Morphological Analyzer (SAMA) (Graff et al., 2009). The morphological analyzer produces a list of possible analyses for each word, and the tagger ranks the mentioned list and selects the optimal full morphological tag for each word in context. MADAMIRA does not expect any diacritics as input and in fact it ignores any naturally occurring diacritics.

In our approach, we use the original full diacritics of the words to filter the ranked choices. Each analysis gets a score based on the edit distance of its associated diacritization with the input diacritization. The analysis with the highest rank and lowest edit distance is selected as the in-context analysis. Since MADAMIRA in default mode produces no analyses sometimes (lexical out-of-vocabulary), we utilize its back off mode to produce all possible analyses (including diacritizations for affixes only).

Diacritization variants While Arabic diacritization conventions are generally well established, we found a number of common stylistic variants in our data sets:

- The long vowel /a:/ is written as ـا *aA* or ا *A* (without the short vowel diacritic).
- The Alif Wasla letter ٱ *Ä* is sometimes written as a bare Alif ا *A*.
- The Alif Hamza Below إ *Ăi* is sometimes written without the short vowel diacritic إ *Ă*.
- The word-final Alif - Fatha nunation character pair can appear as اً *ãA* or ـاً *Aã*.
- The word-final Alif Maqsura - Fatha nunation character pair can appear as ىً *ãý* or ـىً *ýã*.

POS	Diac	Gloss
PV+PVSUFF_SUBJ:3MS	bay∼ana	He demonstrated
PV+PVSUFF_SUBJ:3FP	bay∼an∼a	They demonstrated (f.p)
NOUN_PROP	biyn	Ben
ADJ	bay∼in	Clear
PREP	bayn	Between, among

Table 1: Analyses produced by the morphological analyzer for the word بين *byn*.

- The Sukun diacritic, indicating absence of vowelization can be written or omitted even with (almost) full diacritization.

To make our approach independent of these variants, we normalized them to one form in the edit-distance calculation and the system evaluations. In the above mentioned example pairs, we always normalized to the second variant. As for the Sukun, we always dropped it.

5. Evaluation

In this section, we present our experimental setup, evaluation metrics and experimental results.

5.1. Experimental Setup

Morphological tagger For our baseline system, we use the MADAMIRA morphological tagger (Pasha et al., 2014), which was trained on the training portion of the Penn Arabic Treebank (PATB, parts 1, 2 and 3) (Maamouri et al., 2004; Maamouri et al., 2006; Maamouri et al., 2009) along the recommendations of Diab et al. (2013).

Used texts We selected three fully diacritized and morphologically annotated texts from three genres to report on.

- A newswire genre text (Newswire) is selected from the PATB and includes all of the development set (about 63K words) along the recommendations of Diab et al. (2013). The morphological annotation was done by the LDC and includes all morphological features.
- A children's novel text (Novel) is selected to cover the first chapter (1,175 words) of a book titled وزة السلطان *wzħ AlslTAn* 'The Sultan's Goose' (Kilany, 2013). The morphological annotation was done by the authors and only included diacritization, lemma, POS and tokenization.
- An Islamic jurisprudence text (Religious) is selected to cover the first four pages (970 words) of a book titled مجمع الضمَانَات *mjmς AlDmAnAt* 'Congregation of Guarantees' (Baghdadi, 1987). The morphological annotation was done by the authors and only included diacritization, lemma, POS and tokenization.

The choice of the different genres is intended to measure the effect of similarity between the training data of the baseline morphological tagger (Newswire) and other genres. In later sections we refer to experiments involving Newswire as *in-genre* since the text matches the genre of the morphological tagger; and we accordingly refer to the non-newswire genre experiments as *out-of-genre*.

5.2. Evaluation Metrics

The evaluation was conducted across several accuracy metrics (all on the word level):

- **Diac**: Percentage of words where the chosen analysis has the correct fully diacritized form (with exact spelling).
- **Lex**: Percentage of words where the chosen analysis has the correct lemma.
- **POS**: Percentage of words where the chosen analysis has the correct part-of-speech.
- **Tok**: Percentage of words where the chosen analysis has the correct tokenization.
- **Diac+Lex+POS+Tok**: Percentage of words where the chosen analysis has the correct diacritization, lemma, part-of-speech, and tokenization combined.
- **Star**: Percentage of words where the chosen analysis is perfectly correct (that is, all the morphological features such as gender, number, person, etc. match their gold values). This is the strictest possible metric and is only used for Newswire text.

5.3. Experimental Results

We present next three sets of results on in-genre experiments, out-of-genre experiments and an investigation in the effect of partial diacritization.

5.3.1. In-Genre Experiments

Four experiments were designed to use the baseline system in different ways. One variable was whether to use the tagger system to rank the list of analyses after the analyzer or to use a random ranking instead. To report the results of random ranking, two ways of ranking were performed: alphabetical, and reversed alphabetical, and the average performance was reported. The second variable was the option of using the input diacritization to filter the list of analyses. The filter examines the diacritization of each analysis and keeps only the one(s) with the minimum Levenshtein's edit distance to the input diacritization. The system's output is the top analysis in the final list after ranking and filtering. Table 2. shows the results of the four experiments. The best approach is to use the initial ranking of the morphological tagger and filter the results based on the input diacritization. We achieve 10.2% absolute improvement on the full analysis over the baseline (71% error reduction). Using the

Initial Ranking	Filter	Diac	Lex	POS	Tok	Diac+Lex+POS+Tok	Star
Random	none	35.0	71.6	71.5	79.4	31.8	30.3
Morph. Tagger	none	87.8	96.7	96.5	98.4	86.6	85.6
Random	Full Diac	99.1	96.2	86.2	98.5	85.5	78.6
Morph. Tagger	Full Diac	**99.1**	**98.7**	**97.4**	**99.3**	**97.1**	**95.8**

Table 2: System performance on an in-genre text (63K words of newswire articles).

Genre	System	Diac	Lex	POS	Tok	Diac+Lex+POS+Tok
Newswire	Tagger	87.8	96.7	96.5	98.4	86.6
Newswire	Tagger+Full Diac Filter	**99.1**	**98.7**	**97.4**	**99.3**	**97.1**
Novel	Tagger	82.8	93.0	93.8	96.5	81.0
Novel	Tagger+Full Diac Filter	**98.2**	**97.7**	**96.0**	**99.0**	**95.6**
Religious	Tagger	74.8	89.8	92.3	96.4	72.4
Religious	Tagger+Full Diac Filter	**97.2**	**95.6**	**95.3**	**98.1**	**92.8**

Table 3: System performance on three genres of text.

diacritization filter without the morphological analyzer performs as well as the best system on diacritization, but worse than the morphological analyzer without the filter on other features. A combination of both the morphological analyzer and the diacritization filter is needed to produce high-quality annotations. The annotation scores produced are comparable to reported numbers on inter-annotator agreement and enhanced annotations (Maamouri et al., 2008; Habash and Roth, 2009; Alkuhlani et al., 2013) (although it is hard to make direct apple-to-apple comparisons with the cited efforts).

The best system achieves 93% error reduction on diacritization, 85% on case, 70-80% on voice, mood, state, and the Buckwalter tag, 55-69% on lemma, aspect, person, rationality, and token sequences, 40-49% on gender, number, and enclitics, 18-39% on proclitics, part-of-speech, and gloss. The detailed error reduction for all the features are mentioned in Table 4.

5.3.2. Out-of-Genre Experiments

A comparison between the baseline and the best system was conducted on two extra data sets (of novel, and religious genres), and the results are listed in Table 3. Similar improvements to the previous results can be seen, which suggests that this approach can be used to annotate text of new genres that the morphological tagger was not trained on.

Error analysis We performed a manual investigation of the types of errors in our best system. The error analysis was performed on 1000 words of newswire, and on the full text of the novel and the religious genres. The errors are classified into the following categories according to their order in Table 5.:

- **No (Correct) Analysis**: Words for which no analysis was provided by the morphological analyzer, or no provided analysis was completely correct.
- **Input Error**: Errors caused by typos in the input text.
- **Inter-Word Diac**: Errors caused by phonological epenthesis between words, which is not modeled by the tagger system.
- **POS**: Errors in POS prediction: *Nominals* are noun-adjective errors. *Closed classes* are errors in closed classes of POS, such as particles, pronouns and conjunctions. *Other* are types of POS errors other than the aforementioned ones.
- **Lemma**: Errors in the prediction of the lemma.

It is worth mentioning that we evaluate against a fine-grained POS tag-set which includes 35 tags. Most of POS errors (nominals and closed classes) describe subtle differences within the same class of POS tags, and will disappear when evaluating against a coarser POS tag-set such as CATiB (Habash and Roth, 2009).

Feature	Tagger	Tagger+Filter	Error Reduction
Diacritization	87.8	99.1	92.7
Case	90.5	98.6	85.2
Voice	99.1	99.8	76.6
Buckwalter Tag	85.6	95.9	71.6
Mood	99.1	99.7	70.6
State	96.7	99.0	69.9
Aspect	99.3	99.8	67.8
Lemma	96.7	98.7	62.1
Person	99.2	99.7	62.0
Rationality	99.2	99.7	56.9
Tokenization	98.4	99.3	56.3
Gender	99.3	99.6	49.0
Enclitic 0	99.5	99.8	48.8
Number	99.4	99.6	40.8
Proclitic 1	99.6	99.7	39.1
Proclitic 0	99.6	99.7	31.6
Gloss	92.9	95.1	30.5
POS	96.5	97.4	27.0
Proclitic 3	99.9	99.9	25.0
Proclitic 2	99.6	99.7	17.7
diac+pos+lemma+tok	86.6	97.1	78.6
Star	85.6	95.8	71.1

Table 4: Percentage of error reduction for all features.

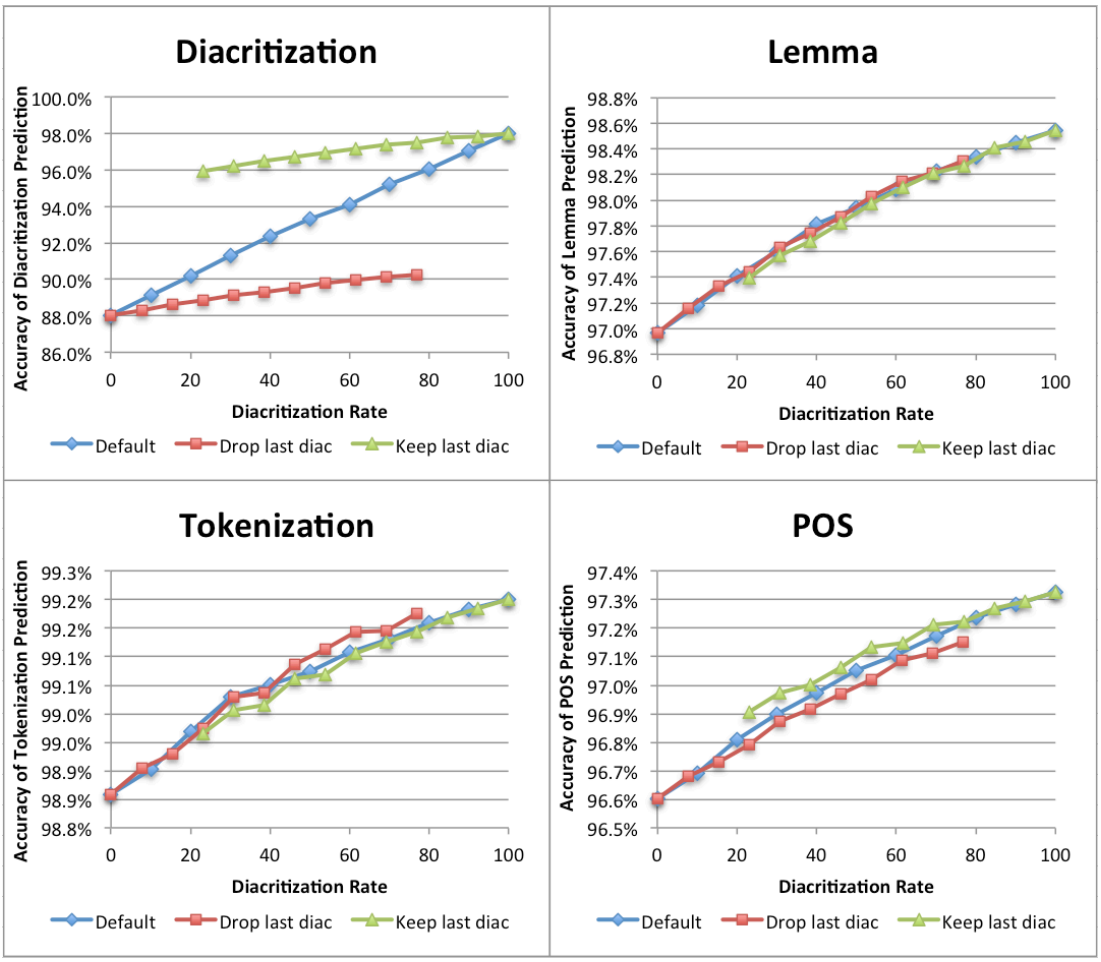

Figure 1: System performance on partially diacritized text.

Error Type	Newswire	Novel	Religious
No Analysis	31.3	5.8	15.7
No Correct Analysis	0	26.9	21.4
Input Error	0	5.8	2.9
Inter-Word Diac	0	3.8	0
POS: Nominals	37.5	28.8	28.6
POS: Closed Classes	6.3	23.1	20.0
POS: Other	6.3	3.8	4.3
Lemma	18.8	1.9	7.1

Table 5: Percentage of errors by type among different genres of text.

5.3.3. Partial Diacritization

In this section we study the effect of using partially diacritized text instead of fully diacritized text. We recognize that using less diacritics is bound to lower the overall quality of the resulting morphological choices; however, we want to understand the nature of the change in quality on different metrics.

Using partially diacritized input instead of fully diacritized input causes a major drop in the performance since the edit-distance measure we use interprets missing diacritics as no-diacritics (or Sukuns) as opposed under-specified diacritics. To address this issue, we created a second mode of the edit distance that does not penalize disagreements involving diacritic underspecification (i.e. missing diacritics) in the input.[3]

Figure 1 shows the performance of our system with the partial-dicritic mode on inputs with different diacritization rates. A linear correlation between the diacritization rate and the performance on all features can be seen. The partially diacritized text is constructed from the fully diacritized form by passing on every diacritic in the text and deciding whether to keep it or not by comparing the output of a random function (in the range of 0-100%) to the diacritization rate. Three settings are illustrated: *Default* is when all diacritics has the same chance of being kept. *Drop last diac* is the same as *Default* except that the last diacritic is always dropped. *Keep last diac* is the same as *Default* except that the last diacritic is always kept. It can be seen from the figure that keeping only the last diacritic (around 23% of total diacritics) gives similar score on predicted dia-

[3]It is worth noting that the partial-diacritic mode performs slightly worse than the full-diacritic mode on fully diacritized text because the fully diacritized gold in fact *does not have a final diacritic* in a small number of cases, which are interpreted as instances of under specification instead of diacritic absence. This results in a drop of 0.6% in the *Star* score for the fully diacritized newswire genre data with partial-diacritic edit distance compared to full-diacritic edit distance.

critization to keeping 80% of the diacritics randomly. This is due to the tagger's weakness at predicting the last diacritic. On lemma, POS, and tokenization, the last diacritic performs in a proportional way to its ratio in the text.
It's worth mentioning that the fully diacritized form contains not only extra diacritics, but also spelling corrections (in around 12% of the words). This is why the partial diacritization system performs better than the baseline even with a 0% diacritization rate (0.3% absolute increase on *Star*). On the other hand, applying the partial diacritization system on a raw text performs worse than the baseline (0.4% absolute decrease on *Star*).

6. Conclusion and Future Work

We have demonstrated a solution to Arabic rich morphological annotation which relies on diacritizations that can be provided by educated Arabic speakers who do not need to be trained in specialized morphological tag sets and guidelines. The solution can work on fully or partially diacritized text. We plan to use this technique to automatically generate large annotated Arabic corpora in less studied genres. The corpora will be diacritized by volunteers and paid typists (potentially using crowd-sourcing). The generated annotations can then be used to enhance Arabic tools.

References

Abandah, G. A., Graves, A., Al-Shagoor, B., Arabiyat, A., Jamour, F., and Al-Taee, M. (2015). Automatic diacritization of Arabic text using recurrent neural networks. *International Journal on Document Analysis and Recognition (IJDAR)*, 18(2):183–197.

Alansary, S. and Nagi, M. (2014). The international corpus of Arabic: Compilation, analysis and evaluation. *ANLP 2014*, page 8.

Alghamdi, M. and Muzafar, Z. (2007). KACST Arabic diacritizer. In *First International Symposium on Computers and Arabic Language*, pages 25–28.

Alkuhlani, S., Habash, N., and Roth, R. (2013). Automatic morphological enrichment of a morphologically underspecified treebank. In *HLT-NAACL*, pages 460–470.

Baghdadi, A. M. i. G. (1987). *Majma' Al-damanat*. Alam Al-Kutub.

Bebah, M., Amine, C., Azzeddine, M., and Abdelhak, L. (2014). Hybrid approaches for automatic vowelization of Arabic texts. *arXiv preprint arXiv:1410.2646*.

Belinkov, Y. and Glass, J. (2015). Arabic diacritization with recurrent neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2281–2285, Lisbon, Portugal, September. Association for Computational Linguistics.

Bouamor, H., Zaghouani, W., Diab, M., Obeid, O., Oflazer, K., Ghoneim, M., and Hawwari, A. (2015). A pilot study on Arabic multi-genre corpus diacritization. In *Proceedings of the Second Workshop on Arabic Natural Language Processing*, pages 80–88, Beijing, China, July. Association for Computational Linguistics.

Buckwalter, T. (2004). Buckwalter Arabic Morphological Analyzer Version 2.0. LDC catalog number LDC2004L02, ISBN 1-58563-324-0.

Diab, M., Habash, N., Rambow, O., and Roth, R. (2013). LDC Arabic treebanks and associated corpora: Data divisions manual. *arXiv preprint arXiv:1309.5652*.

Graff, D., Maamouri, M., Bouziri, B., Krouna, S., Kulick, S., and Buckwalter, T. (2009). Standard Arabic Morphological Analyzer (SAMA) Version 3.1. Linguistic Data Consortium LDC2009E73.

Habash, N. and Rambow, O. (2007). Arabic Diacritization through Full Morphological Tagging. In *Proceedings of the 8th Meeting of the North American Chapter of the Association for Computational Linguistics/Human Language Technologies Conference (HLT-NAACL07)*.

Habash, N. and Roth, R. (2009). CATiB: The Columbia Arabic Treebank. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 221–224, Suntec, Singapore.

Habash, N., Gabbard, R., Rambow, O., Kulick, S., and Marcus, M. (2007a). Determining case in Arabic: Learning complex linguistic behavior requires complex linguistic features. In *Proceeings of the The 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2007)*, Prague, June.

Habash, N., Soudi, A., and Buckwalter, T. (2007b). On Arabic Transliteration. In A. van den Bosch et al., editors, *Arabic Computational Morphology: Knowledge-based and Empirical Methods*. Springer.

Habash, N. (2010). *Introduction to Arabic Natural Language Processing*. Morgan & Claypool Publishers.

Hadj Ameur, M. S., Moulahoum, Y., and Guessoum, A. (2015). Restoration of Arabic diacritics using a multilevel statistical model. In *Computer Science and Its Applications*, volume 456 of *IFIP Advances in Information and Communication Technology*, pages 181–192. Springer International Publishing.

Kilany, K. (2013). *Wazzat Al-sultan*. Hindawi Foundation for Education and Culture.

Maamouri, M. and Cieri, C. (2002). Resources for Natural Language Processing at the Linguistic Data Consortium. In *Proceedings of the International Symposium on Processing of Arabic*, pages 125–146, Manouba, Tunisia.

Maamouri, M., Bies, A., Buckwalter, T., and Mekki, W. (2004). The Penn Arabic Treebank: Building a Large-Scale Annotated Arabic Corpus. In *NEMLAR Conference on Arabic Language Resources and Tools*, pages 102–109, Cairo, Egypt.

Maamouri, M., Bies, A., and Kulick, S. (2006). Diacritization: A challenge to Arabic treebank annotation and parsing. In *Proceedings of the Conference of the Machine Translation SIG of the British Computer Society*.

Maamouri, M., Bies, A., and Kulick, S. (2008). Enhancing the Arabic Treebank: a Collaborative Effort toward New Annotation Guidelines. In European Language Resources Association (ELRA), editor, *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, May.

Maamouri, M., Bies, A., and Kulick, S. (2009). Creating a Methodology for Large-Scale Correction of Treebank Annotation: The Case of the Arabic Treebank. In *Pro-*

ceedings of MEDAR International Conference on Arabic Language Resources and Tools, Cairo, Egypt.

Nelken, R. and Shieber, S. (2005). Arabic Diacritization Using Weighted Finite-State Transducers. In *Proceedings of the Workshop on Computational Approaches to Semitic Languages at 43rd Meeting of the Association for Computational Linguistics (ACL'05)*, pages 79–86, Ann Arbor, Michigan.

Pasha, A., Al-Badrashiny, M., Kholy, A. E., Eskander, R., Diab, M., Habash, N., Pooleery, M., Rambow, O., and Roth, R. (2014). MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic. In *In Proceedings of LREC*, Reykjavik, Iceland.

Rashwan, M., Al-Badrashiny, M., Attia, M., and Abdou, S. (2009). A hybrid system for automatic Arabic diacritization. In *The 2nd International Conference on Arabic Language Resources and Tools*. Citeseer.

Shahrour, A., Khalifa, S., and Habash, N. (2015). Improving Arabic diacritization through syntactic analysis. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 1309–1315.

Smrž, O. and Hajič, J. (2006). The Other Arabic Treebank: Prague Dependencies and Functions. In Ali Farghaly, editor, *Arabic Computational Linguistics: Current Implementations*. CSLI Publications.

Vergyri, D. and Kirchhoff, K. (2004). Automatic Diacritization of Arabic for Acoustic Modeling in Speech Recognition. In Ali Farghaly et al., editors, *COLING 2004 Workshop on Computational Approaches to Arabic Script-based Languages*, pages 66–73, Geneva, Switzerland.

Zitouni, I., Sorensen, J. S., and Sarikaya, R. (2006). Maximum entropy based restoration of Arabic diacritics. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 577–584, Sydney, Australia, July. Association for Computational Linguistics.