Diacritization of Moroccan and Tunisian Arabic Dialects: A CRF Approach

Kareem Darwish*, **Ahmed Abdelali***, **Hamdy Mubarak***, **Younes Samih**†, **Mohammed Attia***
*QCRI, †University of Dusseldorf, *Google Inc.
{kdarwish, aabdelali, hmubarak}@qf.edu.qa, samih@phil.hhu.de, attia@google.com

Abstract

Arabic is written as a sequence of consonants and long vowels, with short vowels normally omitted. Diacritization attempts to recover short vowels and is an essential step for Text-to-Speech (TTS) systems. Though Automatic diacritization of Modern Standard Arabic (MSA) has received significant attention, limited research has been conducted on dialectal Arabic (DA) diacritization. Phonemic patterns of DA vary greatly from MSA and even from one another, which accounts for the noted difficulty of mutual intelligibility between dialects. In this paper we present our research and benchmark results on the automatic diacritization of two Maghrebi sub-dialects, namely Tunisian and Moroccan, using Conditional Random Fields (CRF). Aside from using character n-grams as features, we also employ character-level Brown clusters, which are hierarchical clusters of characters based on the contexts in which they appear. We achieved word-level diacritization errors of 2.9% and 3.8% for Moroccan and Tunisian respectively. We also show that effective diacritization can be performed out-of-context for both sub-dialects.

Keywords: Arabic, Dialects, Vowelization, Diacritization

1. Introduction

Different varieties of Arabic are typically written without diacritics (short vowels). Arabic readers disambiguate words in context and mentally restore diacritics to pronounce words correctly. For Modern Standard Arabic (MSA), diacritics serve dual functions. While word-internal diacritics are phonemic in nature and dictate correct pronunciation and lexical choice, final vowels on words (a.k.a case endings) indicate syntactic role. However, dialects overwhelming use *sukun* as a neutral case-ending for all words, eliminating the need for disambiguating syntactic roles. Thus, dialectal diacritic recovery involves restoring internal-word diacritics only. The task of diacritic restoration is crucial for applications such as text-to-speech (TTS) to enable the proper pronunciation of words.

In this paper, we present new state-of-the-art Arabic diacritization of two sub-dialects of Maghrebi, namely Moroccan and Tunisian, using Conditional Random Fields (CRF) sequence labeling. We trained our CRF sequence labeler using character n-grams as well as character-level Brown clusters. In our context, Brown clusters would bin together characters that appear in similar contexts, which would improve the generalization of the training set. We explore mono-dialectal training as well as cross-dialectal and joint training. Using mono-dialectal training, we achieve word error rates of 2.9% for Moroccan and 3.8% for Tunisian. Though both sub-dialects are orthographically similar, we show that cross-dialectal and joint training lead to significant increases in diacritization errors due to the phonetic divergence of the sub-dialects. Thus, dialectal TTS needs to be tuned for specific sub-dialects.

The contributions of this work are:
• To the best of our knowledge, this is the first work on the diacritization of Maghrebi Arabic, which helps shed more light on the properties of some spoken variants of Arabic and providing benchmark results.
• We show that diacritization can be performed with high accuracy for words out of context.
• We explore the use of cross dialect and joint dialect training between Moroccan and Tunisian, highlighting the orthographic and phonetic similarities and dissimilarities of both sub-dialects.

2. Background

Significant research has addressed diacritic restoration/recovery or diacritization for Arabic, mostly MSA, and some other Semitic languages which are typically written without short vowels. Diacritization is essential for a variety of applications such as TTS and language learning. MSA diacritization involves internal-word diacritization to disambiguate meaning and case ending recovery based on syntactic role. Recovering the case ending is typically significantly harder than core word diacritization. Dialects have mostly eliminated case endings, using the silence diacritic *sukun* instead. Many approaches have been used for internal-word diacritization of MSA such as Hidden Markov Models (Gal, 2002; Darwish et al., 2017), finite state transducers (Nelken and Shieber, 2005), character-based maximum entropy based classification (Zitouni et al., 2006), and deep learning (Abandah et al., 2015; Belinkov and Glass, 2015; Rashwan et al., 2015). Darwish et al. (2017) compared their system to others on common test set. They achieved a word error rate of 3.29% compared 3.04% for Rashwan et al. (2015), 6.73% for Habash and Rambow (2007), and 14.87 for Belinkov and Glass (2015). Azmi and Almajed (2015) survey much of the literature on MSA diacritization.

Concerning dialectal diacritization, the literature is rather scant. Habash et al. (2012) developed a morphological analyzer for dialectal Egyptian, which uses a finite state transducer that encodes manually crafted rules. They report an overall analysis accuracy of 92.1% without reporting diacritization accuracy specifically. Khalifa et al. (2017) developed a morphological analyzer for dialectal Gulf verbs, which also attempts to recover diacritics. Again, they did not specifically report diacritization accuracy. Jarrar et al. (2017) annotated a corpus of dialectal Palestinian containing 43k words. Annotation included text diacritization. In the aforementioned papers, the authors used CODA, a standardized dialectal spelling convention. Other recent work on dialects attempted to perform different processing, such as segmentation, without performing any spelling standardization (Eldesouki et al., 2017; Samih et al., 2017). Diacritization without standardizing spelling is highly advantageous, and thus we pursue character level models in this paper.

3. Data

We were able to obtain two translations of the New Testament into two Maghrebi sub-dialects, namely Moroccan[1] and Tunisian[2] dialects. Both of them are fully diacritized and contain 8,200 verses each. Table 1 shows the data size, and Table 2 gives a sample verse from both dialects, MSA, and the English translation. We split the data for 5-fold cross validation, where training splits were further split 70/10 for training/validation. Given the training portions of each split, Figure 1 shows the distribution of the number of observed diacritized forms per word. As shown, 89% and 82% of words have one diacritized form for Moroccan and Tunisian respectively. We further analyzed the words with more than one form. The percentage of words where one form was used more than 99% of time was 53.8% and 55.5% for Moroccan and Tunisia respectively. We looked at alternative diacritized forms for this group and we found that the less common alternatives are cases were default diacritics (ex. *fatha* before *alef* – روما (rwmA) vs. رُومَا (ruwmaA) – meaning “Rome”)[3] are dropped while they are generally present. Similarly, the percentage of words where the most frequent form was used less than 70% was 6.1% and 8.5% for Moroccan and Tunisian respectively. Aside from the cases where a surface form can have multiple possible diacritics (ex. الْحْكَامْ (AloHokaAmo – “the judging”) – vs. الْحُكَّامْ (AloHuk∼aAmo – “the rulers”)), we found frequent cases where a dicritized form has a *shadda–sukun* combination and another has just *sukun* (ex. يْخَرّْجُو (yoxar∼ojuw) vs. يْخَرْجُو (yoxarojuw) – “to drive out”) and others were different diacritized forms would have nearly identical pronunciation (ex. يْرِيبْ (yoriybo) vs. يْرَيّْبْ (yoray∼obo) – “to destroy”). Further, we used the most frequent diacritized form for each word, and we automatically diacritized the training set. Doing so, the word error rate on the training set was 0.9% and 1.1% for Moroccan and Tunisian respectively. This indicates that diacritizing words out of context can achieve up to 99% accuracy (1% word error rate). We compared this to the MSA version of the same Bible verses (132,813 words) and a subset of diacrtized MSA news articles of comparable size (143,842 words) after removing case-endings. As Table 3 shows, MSA words, particularly for the Bible, have many more possible diacritized forms, and picking the most frequent diacritized form leads to significantly higher word error rate compared to dialects.

Dialects	No. of Words
Moroccan	134,324
Tunisian	131,923

Table 1: Dialectal data size

Lang.	Verse (Colossians 3:20)
Moroccan	آ الْوْلَادْ، طِيعُو وَالِدِيكُمْ فْكُلّْشِي
Tunisian	يَا الوْلَادْ، طِيعُوا وَالْدِيكُمْ فِي كُلْ شَيْء
MSA	أَيُّهَا آلْأَوْلَادُ، أَطِيعُوا وَالِدِيكُمْ فِي آلرَّبِّ
English	Children, obey your parents in all things

Table 2: Sample verse from diacritized dialectal Bibles

We compared the overlap between training and test splits. Figure 2 shows that a little over 93% of the test words were observed during training. If we use the most frequent diacritized forms observed in training, we can diacritize 92.8% and 92.0% of Moroccan and Tunisian words respectively. Thus, the job of a diacritizer is primarily to diacritize words previously unseen words, rather than to disambiguate between different forms. We also compared the cross coverage between the Moroccan and Tunisian datasets. As Figure 3 shows, the overlap is approximately 61%, and the diacritized form in one dialect matches that of the other dialect less than two thirds of the time. This suggests that cross dialect training will yield suboptimal results.

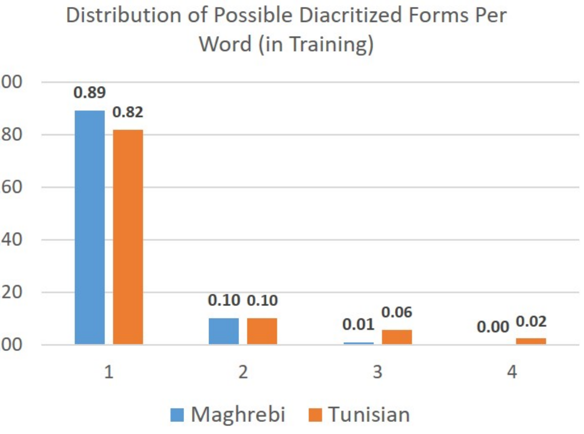

Figure 1: Distribution of the number of diacritized forms per word in training parts

There are 14 diacritics in MSA that Arabic letters can carry[4] in addition to *EMPTY* diacritic which is used for long vowels and sometimes for cases like the definite determiner ال (meaning “the”). In Moroccan, an extra diacritic is also used, namely *shaddah–sukun*. The distributions of different diacritics in Moroccan, Tunisian, and the corresponding MSA of the Bible data are shown in Figure

[1] Translated by Morocco Bible Society

[2] Translated by United Bible Societies, UK

[3] We use Buckwalter encoding to transliterate Arabic words.

[4] https://en.wikipedia.org/wiki/Arabic_script_in_Unicode

	Bible	News
Most Freq	92.1	92.8
No. of Seen Forms		
1	51.7	69.0
2	20.4	26.8
3	13.5	2.9
4	7.1	1.1
≥5	7.3	0.1

Table 3: Distribution of the number of dicaritized forms per word for MSA

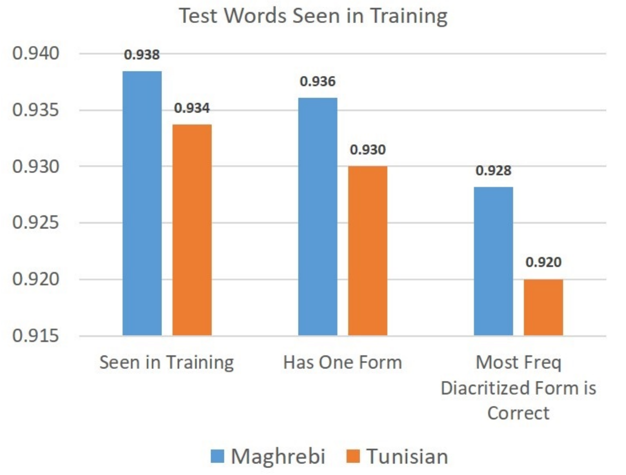

Figure 2: Overlap between train and test parts.

4. Generally, both Moroccan and Tunisian have comparable distributions, and they are different than of MSA. Also, while 34% and 26% of the letters have *sukun* in Moroccan and Tunisian respectively, only 4% of letters in MSA have *sukun*.

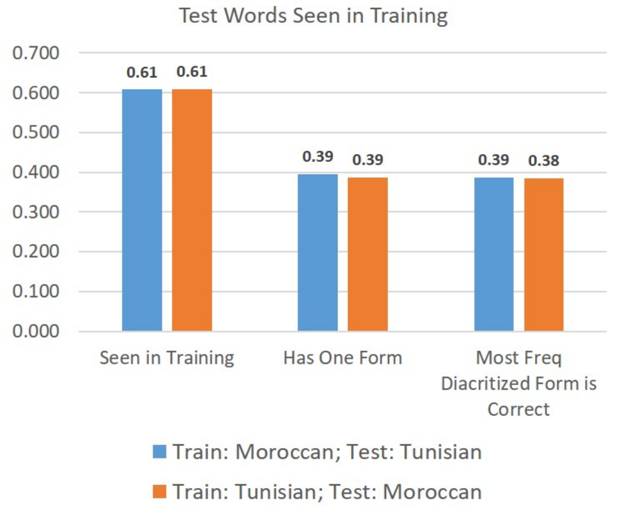

Figure 3: Overlap between train and test parts.

Figures 5 and 6 show distributions of diacritics for first letter (prompt) and last letter (typically case ending indicating grammatical function) in words to show how diacritization of Moroccan and Tunisian differ from MSA. In MSA, words cannot start with a letter with *sukun* because there is no morphological templatic pattern that starts with *sukun*. However, in Moroccan and Tunisian, 43% and 23% of the words start with *sukun*. For the last diacritic, MSA case endings can take many values. Conversely, Moroccan and Tunisian case endings are overwhelmingly either *sukun* (57% and 53%) or *EMPTY* (37% and 45%) respectively.

It is worth mentioning that in our corpus, the maximum number of *sukun* in a word is 6 for both dialects with words like وْلْلْبْلَايْصْ (wololobolaAyoSo – “and places”) and وْنَبْعْثُوهُمْلْكُمْ (wonaboEovuhumolokumo – “and we send them to you”) compared to only a maximum of 3 *sukun* in MSA words like فَأُشْفِيهِمْ (fa>u$ofiyhimo – “I will heal them”). Also, 23% of Moroccan words and 7% of Tunisian words have consecutive *sukun*, and the maximum number of consecutive *sukun* in Moroccan is 5, as in وْفْتْلْثْ (wofotoloto – meaning “and in three”), compared to only 2 for Tunisian, as in تِتْضْرَبْشْ (titoDorabo$o – “will not hit”). In the MSA Bible, there is only one word that has 2 consecutive sukun, namely سِمِيْرْنا (simiyorokA – a foreign named-entity “Smyrna”), because no words of Arabic origin are allowed to have 2 consecutive *sukun*. If two *sukun* happen to appear consecutively, MSA diacritization rules convert the first *sukun* to either *fatha* or *kasra*.

The Word ومشى (wm$Y – “and he walked”) is an example of words that are written the same in Moroccan, Tunisian, and MSA with the same meaning but with different diacritization and hence pronunciation: وْمْشَى (womo$aY) in Moroccan; وِمْشَى (wimo$aY) in Tunisian; and وَمَشَى (wama$aY) in MSA. All the above indicate that using an MSA diacritizer to diacritize Moroccan or Tunisian would lead to high word error rate, because they follow different diacritization patterns and rules.

4. Proposed Approach: Linear Chain CRF

The effectiveness of CRFs (Lafferty et al., 2001) has been shown for many sequence labeling tasks, such as POS tagging and named entity recognition. CRFs effectively combine state-level features with transition features. They are simple and well-understood, and they usually provide efficient models with close to state-of-the-art results. Thus, CRF is a potentially effective method to apply to this task. For all the experiments, we used the CRF++ implementation of a CRF sequence labeler with L2 regularization and default value of 10 for the generalization parameter “C”.[5]

In our setup, our goal is to tag each character of every word with the appropriate diacritic, where character-level diacritics are our labels. For features, given a word of character sequence c_n ... $c_{-2}, c_{-1}, c_0, c_1, c_2$ ... c_m, we used a combination of character n-gram features, namely unigram (c_0), bigrams (c^0_{-1}; c^1_0), trigrams (c^0_{-2}; c^1_{-1}; c^2_0), and 4-grams (c^0_{-3}; c^1_{-2}; c^2_{-1}; c^3_0).

[5] https://github.com/taku910/crfpp

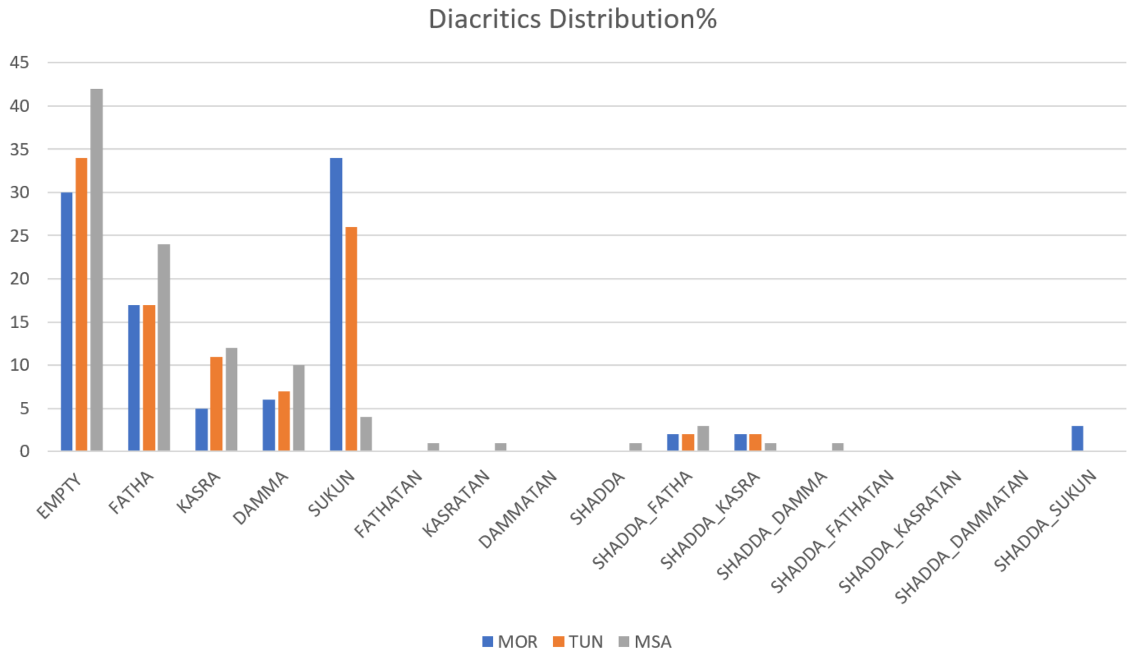

Figure 4: Diacritics Distribution in Moroccan, Tunisian, and MSA

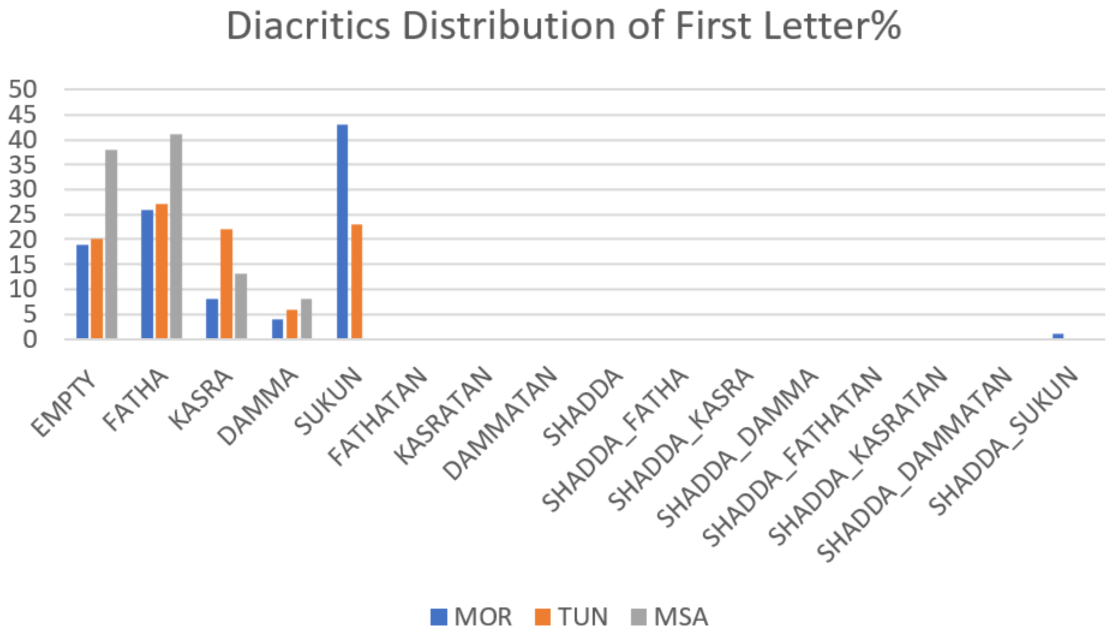

Figure 5: Diacritics Distribution of First Letter

Another feature that may potentially help our sequence labeling to generalize is the use of character level Brown clusters (Brown et al., 1992), which are hierarchical clusters of tokens based on the contexts in which they appear. They have been shown to improve many NLP tasks such as POS tagging (Owoputi et al., 2013). The rationale for using it here is that some characters may appear in similar contexts and would hence have similar diacrtics. The advantage is that Brown clusters can be learned from unlabeled texts. We generated 25 character clusters from the training part for each fold using the implementation of Liang (2005). When using Brown clusters, we used the aforementioned character n-gram features in addition to an identical set of features where we replace characters with their corresponding Brown cluster tags. Given that the vast majority of dialectal words have only one possible diacritized form, the CRF is trained on individual words out of context.

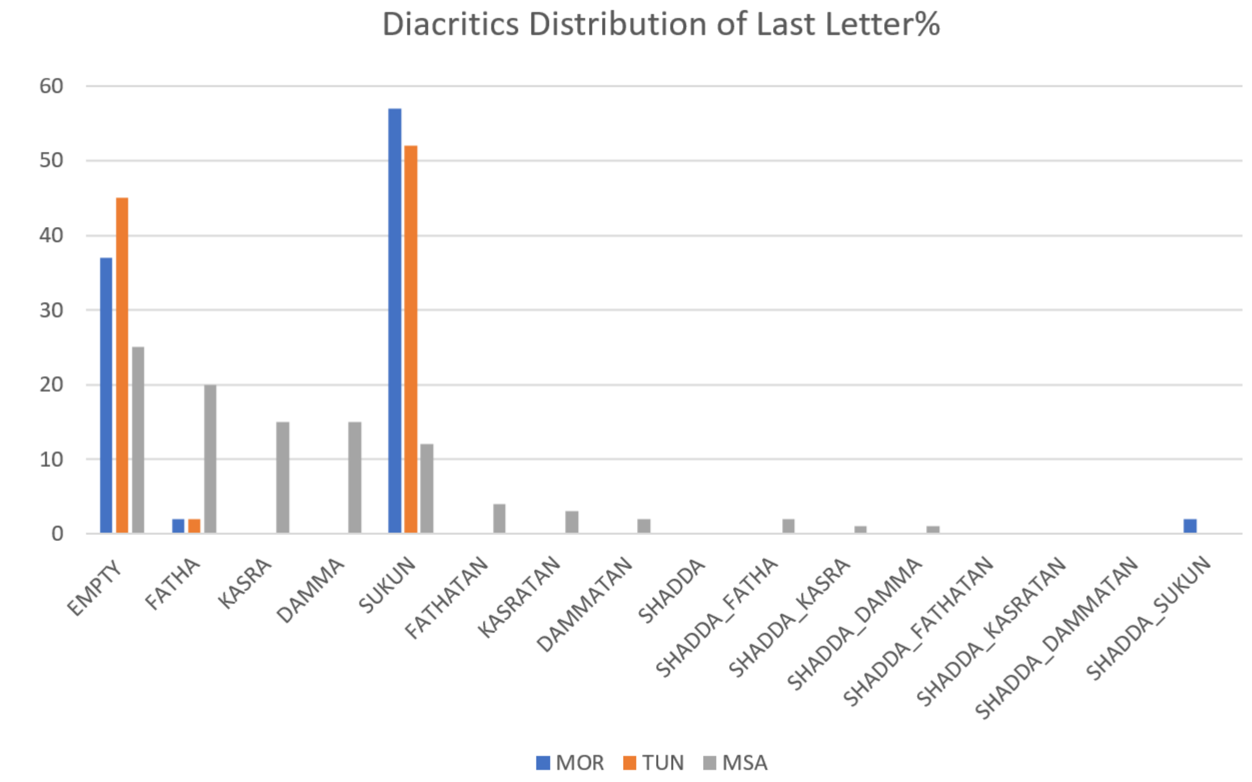

Figure 6: Diacritics Distribution of Last Letter

Training Set	Test Set	Error Rate Character	Word
(a) Uni-dialectal Training			
Moroccan	Moroccan	1.1	3.1
Tunisian	Tunisian	1.8	4.0
(b) Cross Training			
Moroccan	Tunisian	17.2	**43.3**
Tunisian	Moroccan	17.9	**43.9**
(c) Combined Training			
Combined	Moroccan	3.0	8.7
Combined	Tunisian	4.8	13.8

Table 4: CRF character n-grams results – Average across all folds

Training Set	Test Set	Error Rate Character	Word
(a) Uni-dialectal Training			
Moroccan	Moroccan	1.1	2.9
Tunisian	Tunisian	1.7	3.8
(b) Cross Training			
Moroccan	Tunisian	20.1	47.0
Tunisian	Moroccan	20.8	48.9
(c) Combined Training			
Combined	Moroccan	12.6	34.2
Combined	Tunisian	9.5	23.8

Table 5: CRF Results with Brown clusters – Average across all folds

5. Results

As shown in Figure 2, our baseline uses the most frequently seen diacritized form that is observed in training and skips unseen words. Word error rate of the baseline is 7.2% and 8.0% for Moroccan and Tunisian respectively. We conducted three sets of experiments:

First, we trained and tested on the same dialectal data. Table 4 (a) shows that we are able to achieve word error rate of 3.1% and 4.0% for Moroccan and Tunisian respectively. When we used Brown clusters (Table 5 (a)), errors decreased by 0.2% absolute for both dialects. In effect, we are able to properly diacritize 56.9% and 50.0% of unseen words or incorrectly diacritized words by the baseline for both dialects respectively.

Second, we wanted to see if both dialects can learn from each other. As Table 4 (b) shows, we trained on one dialect and tested on the other. Adding Brown clusters (Table 5 (b)) lowered results even further. As expected based on our discussion in Section 3., the results were markedly lower, and improvements in diacritizing one dialect would further degrade cross-dialectal results. This validates the claim that word diacritizations in different sub-dialects dialects are significantly different.

Third, we combined training data from both dialects, and we tested on individual dialects. As Table 4 (c) shows, combining data led to results that are worse than the baseline. Using Brown clusters, as shown in Table 5 (c), made results even worse. This is not surprising given the fact that many words appear in both dialects and are diaritized differently. If both dialects could learn from each other, then perhaps we could have a system that can diacritize either dialect without prior dialect identification. Unfortunately, that is not the case.

6. Discussion and Conclusion

In this paper we presented our work on the diacritization of sub-dialects of Maghrebi Arabic, namely Moroccan and Tunisian. Diacritization is essential for applications such as TTS to properly pronounce words. We noted that dialectal Arabic is less contextual and more predictable than Modern Standard Arabic, and hence high levels of accuracy (low word error rates) can be achieved, to a large extent context free. Using linear chain CRF sequence labeling with character n-grams and character-level Brown clusters, we achieved a word error rate of 2.9% and 3.8% for Moroccan and Tunisian respectively. When we performed cross training the accuracy dropped significantly, which reveals that, even for closely-related dialects, there is a great divergence in pronunciation patterns. For future work, we plan to explore deep learning for diacritization.

7. Bibliographical 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.

Azmi, A. M. and Almajed, R. S. (2015). A survey of automatic arabic diacritization techniques. *Natural Language Engineering*, 21(03):477–495.

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.

Brown, P. F., Desouza, P. V., Mercer, R. L., Pietra, V. J. D., and Lai, J. C. (1992). Class-based n-gram models of natural language. *Computational linguistics*, 18(4):467–479.

Darwish, K., Mubarak, H., and Abdelali, A. (2017). Arabic diacritization: Stats, rules, and hacks. In *Proceedings of the Third Arabic Natural Language Processing Workshop*, pages 9–17.

Eldesouki, M., Samih, Y., Abdelali, A., Attia, M., Mubarak, H., Darwish, K., and Laura, K. (2017). Arabic multi-dialect segmentation: bi-lstm-crf vs. svm. *arXiv preprint arXiv:1708.05891*.

Gal, Y. (2002). An hmm approach to vowel restoration in arabic and hebrew. In *Proceedings of the ACL-02 workshop on Computational approaches to Semitic languages*, pages 1–7. Association for Computational Linguistics.

Habash, N. and Rambow, O. (2007). Arabic diacritization through full morphological tagging. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*, pages 53–56. Association for Computational Linguistics.

Habash, N., Eskander, R., and Hawwari, A. (2012). A morphological analyzer for egyptian arabic. In *Proceedings of the twelfth meeting of the special interest group on computational morphology and phonology*, pages 1–9. Association for Computational Linguistics.

Jarrar, M., Habash, N., Alrimawi, F., Akra, D., and Zalmout, N. (2017). Curras: an annotated corpus for the palestinian arabic dialect. *Language Resources and Evaluation*, 51(3):745–775.

Khalifa, S., Hassan, S., and Habash, N. (2017). A morphological analyzer for gulf arabic verbs. *WANLP 2017 (co-located with EACL 2017)*, page 35.

Lafferty, J., McCallum, A., and Pereira, F. C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Proceedings of the 18th International Conference on Machine Learning 2001 (ICML 2001)*, pages 282–289.

Liang, P. (2005). *Semi-supervised learning for natural language*. Ph.D. thesis, Massachusetts Institute of Technology.

Nelken, R. and Shieber, S. M. (2005). Arabic diacritization using weighted finite-state transducers. In *Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages*, pages 79–86. Association for Computational Linguistics.

Owoputi, O., O'Connor, B., Dyer, C., Gimpel, K., Schneider, N., and Smith, N. A. (2013). Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of NAACL-HLT 2013*, pages 380–390. Association for Computational Linguistics.

Rashwan, M., Al Sallab, A., Raafat, M., and Rafea, A. (2015). Deep learning framework with confused sub-set resolution architecture for automatic arabic diacritization. In *IEEE Transactions on Audio, Speech, and Language Processing*, pages 505–516.

Samih, Y., Eldesouki, M., Attia, M., Darwish, K., Abdelali, A., Mubarak, H., and Kallmeyer, L. (2017). Learning from relatives: Unified dialectal arabic segmentation. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 432–441.

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 the 44th annual meeting of the Association for Computational Linguistics*, pages 577–584. Association for Computational Linguistics.