Learning from Relatives: Unified Dialectal Arabic Segmentation

Younes Samih[1], **Mohamed Eldesouki**[3], **Mohammed Attia**[2], **Kareem Darwish**[3], **Ahmed Abdelali**[3], **Hamdy Mubarak**[3], **and Laura Kallmeyer**[1]

[1]Dept. of Computational Linguistics,University of Düsseldorf, Düsseldorf, Germany
[2]Google Inc., New York City, USA
[3]Qatar Computing Research Institute, HBKU, Doha, Qatar
[1]{samih,kallmeyer}@phil.hhu.de
[2]attia@google.com
[3]{mohamohamed,hmubarak,aabdelali,kdarwish}@hbku.edu.qa

Abstract

Arabic dialects do not just share a common koiné, but there are shared pan-dialectal linguistic phenomena that allow computational models for dialects to learn from each other. In this paper we build a unified segmentation model where the training data for different dialects are combined and a single model is trained. The model yields higher accuracies than dialect-specific models, eliminating the need for dialect identification before segmentation. We also measure the degree of relatedness between four major Arabic dialects by testing how a segmentation model trained on one dialect performs on the other dialects. We found that linguistic relatedness is contingent with geographical proximity. In our experiments we use SVM-based ranking and bi-LSTM-CRF sequence labeling.

1 Introduction

Segmenting Arabic words into their constituent parts is important for a variety of applications such as machine translation, parsing and information retrieval. Though much work has focused on segmenting Modern Standard Arabic (MSA), recent work began to examine dialectal segmentation in some Arabic dialects. Dialectal segmentation is becoming increasingly important due to the ubiquity of social media, where users typically write in their own dialects as opposed to MSA. Dialectal text poses interesting challenges such as lack of spelling standards, pervasiveness of word merging, letter substitution or deletion, and foreign word borrowing. Existing work on dialectal segmentation focused on building resources and tools for each dialect separately (Habash et al., 2013; Pasha et al., 2014; Samih et al., 2017). The rational for the separation is that different dialects have different affixes, make different lexical choices, and are influenced by different foreign languages. However, performing reliable dialect identification to properly route text to the appropriate segmenter may be problematic, because conventional dialectal identification may lead to results that are lower than 90% (Darwish et al., 2014). Thus, building a segmenter that performs reliably across multiple dialects without the need for dialect identification is desirable.

In this paper we examine the effectiveness of using a segmenter built for one dialect in segmenting other dialects. Next, we explore combining training data for different dialects in building a joint segmentation model for all dialects. We show that the joint segmentation model matches or outperforms dialect-specific segmentation models. For this work, we use training data in four different dialects, namely Egyptian (EGY), Levantine (LEV), Gulf (GLF), and Maghrebi (MGR). We utilize two methods for segmentation. The first poses segmentation as a ranking problem, where we use an SVM ranker. The second poses the problem as a sequence labeling problem, where we use a bidirectional Long Short-Term Memory (bi-LSTM) Recurrent Neural Network (RNN) that is coupled with Conditional Random Fields (CRF) sequence labeler.

2 Background

Work on dialectal Arabic is fairly recent compared to MSA. A number of research projects were devoted to dialect identification (Biadsy et al., 2009; Zbib et al., 2012; Zaidan and Callison-Burch, 2014; Eldesouki et al., 2016). There are five major dialects including Egyptian, Gulf, Iraqi, Levantine and Maghrebi. Few resources for these dialects

Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017), pages 432–441,
Vancouver, Canada, August 3 - August 4, 2017. ©2017 Association for Computational Linguistics

are available such as the CALLHOME Egyptian Arabic Transcripts (LDC97T19), which was made available for research as early as 1997. Newly developed resources include the corpus developed by Bouamor et al. (2014), which contains 2,000 parallel sentences in multiple dialects and MSA as well as English translation. These sentences were translated by native speakers into the target dialects from an original dialect, the Egyptian.

For segmentation, Mohamed et al. (2012) built a segmenter based on memory-based learning. The segmenter has been trained on a small corpus of Egyptian Arabic comprising 320 comments containing 20,022 words from www.masrawy.com that were segmented and annotated by two native speakers. They reported a 91.90% accuracy on the segmentation task. MADA-ARZ (Habash et al., 2013) is an Egyptian Arabic extension of the Morphological Analysis and Disambiguation of Arabic (MADA) tool. They trained and evaluated their system on both Penn Arabic Treebank (PATB) (parts 1-3) and the Egyptian Arabic Treebank (parts 1-5) (Maamouri et al., 2014) and they achieved 97.5% accuracy. MADAMIRA[1] (Pasha et al., 2014) is a new version of MADA that includes the functionality for analyzing dialectal Egyptian. Monroe et al. (2014) used a single dialect-independent model for segmenting Egyptian dialect in addition to MSA. They argue that their segmenter is better than other segmenters that use sophisticated linguistic analysis. They evaluated their model on three corpora, namely parts 1-3 of Penn Arabic Treebank (PATB), Broadcast News Arabic Treebank (BN), and parts 1-8 of the BOLT Phase 1 Egyptian Arabic Treebank (ARZ) reporting an F1 score of 92.1%.

3 Segmentation Datasets

We used datasets for four dialects, namely Egyptian (EGY), Levantine (LEV), Gulf (GLF), and Maghrebi (MGR) which are available at http://alt.qcri.org/resources/da_resources/. Each dataset consists of a sets of 350 manually segmented tweets. Briefly, we obtained a large Arabic collection composed of 175 million Arabic tweets by querying the Twitter API using the query "lang:ar" during March 2014. Then, we identified tweets whose authors identified their location in countries where the dialects of interest are spoken (e.g. Morocco, Algeria, Tunisia, and Libya for MGR) using a large location gazetteer (Mubarak and Darwish, 2014) which maps each region/city to its country. Then we filtered the tweets using a list containing 10 strong dialectal words per dialect, such as the MGR word كيما "kymA" (like/as in) and the LEV word هيك "hyk" (like this). Given the filtered tweets, we randomly selected 2,000 unique tweets for each dialect, and we asked a native speaker of each dialect to manually select 350 tweets that are heavily dialectal, i.e. contain more dialectal than MSA words. Table 1 lists the number of tweets that we obtained for each dialect and the number of words they contain.

Dialect	No of Tweets	No of Tokens
Egyptian	350	6,721
Levantine	350	6,648
Gulf	350	6,844
Maghrebi	350	5,495

Table 1: Dataset size for the different dialects

We manually segmented each word in the corpus while preserving the original characters. This decision was made to allow processing real dialectal words in their original form. Table 2 shows segmented examples from the different dialects.

3.1 Segmentation Convention

In some research projects, segmentation of DA is done on a CODA'fied version of the text, where CODA is a standardized writing convention for DA (Habash et al., 2012). CODA guidelines provide directions on to how to normalize words, correct spelling and unify writing. Nonetheless, these guidelines are not available for all dialects. In the absence of such guidelines as well as the dynamic nature of the language, we choose to operate directly on the raw text. As in contrast to MSA, where guidelines for spelling are common and standardized, written DA seems to exhibit a lot of diversity, and hence, segmentation systems need to be robust enough to handle all the variants that might be encountered in such texts.

Our segmentation convention is closer to stemming rather than tokenization in that we separate all prefixes (with the exception of imperfective prefixes with verbs) and suffixes from the stems. The following is a summary to these instructions that were given to the native speakers to segment the data:

[1] MADAMIRA release 20160516 2.1

Word	Glossary	Segmentation	Dialect
بيقولك “byqwlk”	Is telling you	ب+يقول+ك “b+yqwl+k”	EGY
و يجي “wyjy”	And he comes	و+يج+ي “w+yj+y”	GLF
برد “brd”	I’ll return	ب+رد “b+rd”	LEV
مغتنفاعهم “mgtnfAEhm”	It will not benefit them	م+غ+تنفاع+هم “m+g+tnfAE+hm”	MGR

Table 2: Dialect annotation example

- Separate all prefixes for verbs, nouns, and adjectives, e.g. the conjunction و “w” (and), preposition ل “l” (to), definite article ال “Al” (the), etc.
- Separate all suffixes for verbs, nouns, and adjectives, e.g. the feminine marker ة “p”, number marker ون “wn”, object or genitive pronouns ه “h” (him), etc.
- Emoticons, user names, and hash-tags are treated as single units.
- Merged words are separated, e.g. عبد+ال+عزيز “Ebd+Al+Ezyz” (Abd Al-Aziz).
- When there is an elongation of a short vowel “a, u ,i” with a preposition, the elongated vowel is segmented with the preposition, e.g. ليهم “lyhm” (for them) ⇒ لي+هم “ly+hm”.

Complete list of guidelines is found at: http://alt.qcri.org/resources/da_resources/seg-guidelines.pdf.

4 Arabic Dialects

4.1 Similarities

There are some interesting observations which show similar behavior of different Arabic dialects (particularly those in our dataset) when they diverge from MSA. These observations show that Arabic dialects do not just share commonalities with MSA, but they also share commonalities among themselves. It seems that dialects share some built-in functionalities to generate words, some of which may have been inherited from classical Arabic, where some of these functionalities are lost or severely diminished in MSA. Some of these commonalities include:

- Dialects have eliminated case endings.
- Dialects introduce a progressive particle, e.g. ب+يقول “b+yqwl” (EGY), عم+يقول “Em+yqwl” (LEV), ك+يقول “k+yqwl” (MGR), and د+يقول “d+yqwl” (Iraqi) for “he says”. This does not exist in MSA.
- Some dialects use a post-negation particle, e.g. م+يحب+ش “m+yHb+$” (does not like) (EGY, LEV and MGR). This does not also exist in MSA as well as GLF.
- Dialects have future particles that are different from MSA, such as ح “H” (LEV), ه “h” (EGY), and غ “g” (MGR). Similar to the MSA future particle س “s” that may have resulted from shortening the particle سوف “swf” and then using the shortened version as a prefix, dialectal future particles may have arisen using a similar process, where the Levantine future particle “H” is a shortened version of the word راح “rAH” (he will) (Persson, 2008; Jarad, 2014).
- Dialects routinely employ word merging, particularly when two identical letters appear consecutively. In MSA, this is mostly restricted to the case of the preposition ل “l” (to) when followed by the determiner ال “Al” (the), where the “A” in the determiner is silent. This is far more common in dialects as in يعمل لك “yEml lk” (he does for you) ⇒ يعملك “yEmlk”.
- Dialects often change short vowels to long vowels or vice verse (vowel elongation and reduction). This phenomenon infrequently appears in poetry, particularly classical Arabic poetry, but is quite common in dialects such as converting له “lh” (to him) to ليه “lyh”.
- Dialects have mostly eliminated dual forms except with nouns, e.g. عيني “Eyny” (my two eyes) and قرشين “qr$yn” (two piasters). Consequently dual agreement markers on adjectives, relative pronouns, demonstrative adjectives, and

verbs have largely disappeared. Likewise, masculine nominative plural noun and verb suffix ون "wn" has been largely replaced with the accusative/genitive forms ين "yn" and وا "wA" respectively.

Phenomena that appear in multiple dialects, but may not necessarily appear in MSA, may provide an indication that segmented training data for one dialect may be useful in segmenting other dialects.

4.2 Differences

In this section, we show some differences between dialects that cover surface lexical and morphological features in light of our datasets. Deep lexical and morphological analysis can be applied after POS-tagging of these datasets. Differences can explain why some dialects are more difficult than others, which dialects are closer to each other, and the possible effect of cross-dialect training. The differences may also aid future work on dialect identification.

We start by comparing dialects with MSA to show how close a dialect to MSA is. We randomly selected 300 words from each dialect and we analyzed them using the Buckwalter MSA morphological analyzer (BAMA) (Buckwalter, 2004). Table 3 lists the percentage of words that were analyzed, analysis precision, and analysis recall, which is the percentage of actual MSA words that BAMA was able to analyze. Results show that BAMA was most successful, in terms of coverage and precision, in analyzing GLF, while it faired the worst on MGR, in terms of coverage, and the worst on LEV, in terms of precision. Some dialectal words are incorrectly recognized as MSA by BAMA, such as كده "kdh" (like this), where BAMA analyzed it as "kd+h" (his toil). It seems that GLF is the closest to MSA and MGR is the furthest away.

Dialect	Percent Analyzed	Analysis Precision	Analysis Recall
EGY	83	81	94
LEV	83	76	91
GLF	**86**	**88**	94
MGR	78	78	95

Table 3: Buckwalter analysis

Table 4 shows the overlap between unique words and all words for the different dialect pairs in our datasets. As the table shows, EGY, LEV, and GLF are closer together and MGR is further away from all of them. Also, LEV is closer to both EGY and GLF than the last two to each other. We also looked at the common words between dialects to see if they had different segmentations. Aside from two words, namely ليه "lyh" (to him, why) and بيه "byh" (with it, gentleman), that both appear in EGY and LEV, all other common words have identical segmentations. This is welcome news for the lookup scheme that we employ in which we use segmentations that are seen in training directly during testing.

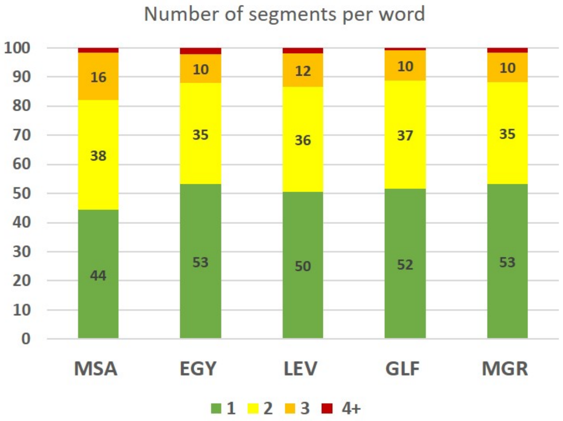

Figure 1: Distribution of segment count per word (percentages are overlaid on the graph)

Dialect pairs	Unique Overlap	All Overlap
EGY-GLF	16.1%	41.6%
EGY-LEV	18.1%	43.3%
EGY-MGR	14.3%	36.7%
GLF-LEV	17.0%	41.4%
GLF-MGR	15.9%	37.8%
LEV-MGR	16.2%	38.5%

Table 4: Common words across dialects

Figure 1 shows the distribution of segment counts per word for words in our datasets. We obtained the MSA segment counts from the Arabic Penn Treebank (parts 1-3) (Maamouri et al., 2014). The figure shows that dialectal words tend to have a similar distribution of word segment counts and they generally have fewer segments than MSA. This may indicate that dialects may have simpler segmentations than MSA, and cases where words have 4 or more segments, such as

م+قلت+ها+ل+و+ش "m+qlt+hA+l+w+$" (I did not say it to him), are infrequent.

Tables 5 and 6 respectively show the number of prefixes or suffixes, the top 5 prefixes and suffixes (listed in descending order), and the unique prefixes and suffixes for each dialect in comparison to MSA. As the tables show, MGR has the most number of prefixes, while GLF has the most number of suffixes. Further, there are certain prefixes and suffixes that are unique to dialects. While the prefix "Al" (the) leads the list of prefixes for all dialects, the prefix ب "b" in LEV and EGY, where it is either a progressive particle or a preposition, is used more frequently than in MSA, where it is used strictly as a preposition. Similarly, the suffix كن "kn" (your) is more frequent in LEV than any other dialect. The Negation suffix ش "$" (not) and feminine suffix marker كي "ky" (your) are used in EGY, LEV, and MGR, but not in GLF or MSA. The appearance of some affixes in some dialects and their absence in others may seem to complicate cross dialect training, and the varying frequencies of affixes across dialects may seem to complicate joint training.

Dialect	No.	Top 5	Unique
MSA	8	Al,w,l,b,f	>, s
EGY	11	Al,b,w,m,h	hA, fA
LEV	11	Al,b,w,l,E	Em
GLF	14	Al,w,b,l,mA	mw,mb,$
MGR	**19**	Al,w,l,b,mA	kA,t,tA,g

Table 5: Prefixes statistics

Dialect	No.	Top 5	Unique
MSA	23	p,At,A,h,hA	hmA
EGY	24	h,p,k,$,hA	Y,kwA,nY,kY
LEV	27	p,k,y,h,w	-
GLF	**30**	h,k,y,p,t	j
MGR	24	p,w,y,k,hA	Aw

Table 6: Suffixes statistics

5 Learning Algorithms

We present here two different systems for word segmentation. The first uses SVM-based ranking (SVM^{Rank})[2] to rank different possible segmentations for a word using a variety of features. The second uses bi-LSTM-CRF, which performs character-based sequence-to-sequence mapping to predict word segmentation.

5.1 SVM^{Rank} Approach

We used the SVM-based ranking approach proposed by Abdelali et al. (2016), in which they used SVM based ranking to ascertain the best segmentation for Modern Standard Arabic (MSA), which they show to be fast and of high accuracy. The approach involves generating all possible segmentations of a word and then ranking them. The possible segmentations are generated based on possible prefixes and suffixes that are observed during training. For example, if hypothetically we only had the prefixes و "w" (and) and ل "l" (to) and the suffix ه "h" (his), the possible segmentations of وليده "wlydh" (his new born) would be {wlydh, w+lydh, w+l+ydh, w+l+yd+h, w+lyd+h, wlyd+h} with "wlyd+h" being the correct segmentation. SVM^{Rank} would attempt to rank the correct segmentation higher than all others. To train SVM^{Rank}, we use the following features:

- Conditional probability that a leading character sequence is a prefix.
- Conditional probability that a trailing character sequence is a suffix.
- probability of the prefix given the suffix.
- probability of the suffix given the prefix.
- unigram probability of the stem.
- unigram probability of the stem with first suffix.
- whether a valid stem template can be obtained from the stem, where we used Farasa (Abdelali et al., 2016) to guess the stem template.
- whether the stem that has no trailing suffixes and appears in a gazetteer of person and location names (Abdelali et al., 2016).
- whether the stem is a function word, such as على "ElY" (on) and من "mn" (from).
- whether the stem appears in the AraComLex[3] Arabic lexicon (Attia et al., 2011) or in the Buckwalter lexicon (Buckwalter, 2004). This is sensible considering the large overlap between MSA and DA.
- length difference from the average stem length.

[2] https://www.cs.cornell.edu/people/tj/svm_light/svm_rank.html

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

The segmentations with their corresponding features are then passed to the SVM ranker (Joachims, 2006) for training. Our SVM^{Rank} uses a linear kernel and a trade-off parameter between training error and margin of 100. All segmentations are ranked out of context. Though some words may have multiple valid segmentations in different contexts, previous work on MSA has shown that it holds for 99% of the cases (Abdelali et al., 2016). This assumption allows us to improve segmentation results by looking up segmentations that were observed in the dialectal training sets (DA) or segmentations from the training sets with a back off to segmentation in a large segmented MSA corpus, namely parts 1, 2, and 3 of the Arabic Penn Treebank Maamouri et al. (2014) (DA+MSA).

5.2 Bi-LSTM-CRF Approach

In this subsection we describe the different components of our Arabic segmentation bi-LSTM-CRF based model, shown in Figure 2. It is a slight variant of the bi-LSTM-CRF architecture first proposed by Huang et al. (2015), Lample et al. (2016), and Ma and Hovy (2016)

5.2.1 Recurrent Neural Networks

A recurrent neural network (RNN) together with its variants, i.e. LSTM, bi-LSTM, GRU, belong to a family of powerful neural networks that are well suited for modeling sequential data. Over the last several years, they have achieved many groundbreaking results in many NLP tasks. Theoretically, RNNs can learn long distance dependencies, but in practice they fail due to vanishing/exploding gradients (Bengio et al., 1994).

LSTMs LSTMs (Hochreiter and Schmidhuber, 1997) are variants of the RNNs that can efficiently overcome difficulties with training and efficiently cope with long distance dependencies. Formally, the output of the LSTM hidden layer h_t given input x_t is computed via the following intermediate calculations: (Graves, 2013):

$$\begin{aligned}
i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \\
f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \\
c_t &= f_t c_{t-1} + i_t \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \\
o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \\
h_t &= o_t \tanh(c_t)
\end{aligned}$$

where σ is the logistic sigmoid function, and i, f, o and c are respectively the input gate, forget gate, output gate and cell activation vectors. More interpretation about this architecture can be found in (Graves and Schmidhuber, 2005) and(Lipton et al., 2015).

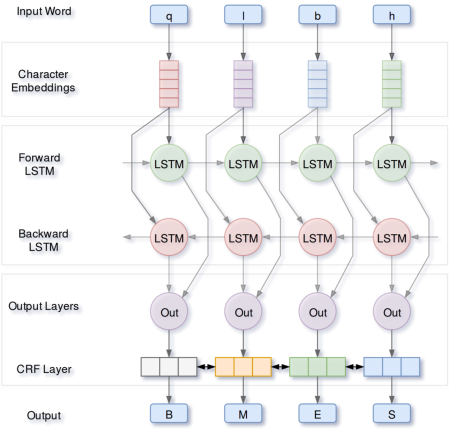

Figure 2: Architecture of our proposed neural network Arabic segmentation model applied to the word قلبه "qlbh" and output "qlb+h".

Bi-LSTMs Another extension to the single LSTM networks are the bi-LSTMs (Schuster and Paliwal, 1997). They are also capable of learning long-term dependencies and maintain contextual features from both past and future states. As shown in Figure 2, they are comprised of two separate hidden layers that feed forwards to the same output layer.

CRF In many sequence labeling tasks bi-LSTMs achieve very competitive results against traditional models, still when they are used for some specific sequence classification tasks, such as segmentation and named entity detection, where there is a strict dependence between the output labels, they fail to generalize perfectly. During the training phase of the bi-LSTM networks, the resulting probability distribution of each time step is independent from each other. To overcome this independence assumptions imposed by the bi-LSTM and to exploit this kind of labeling constraints in our Arabic segmentation system, we model label sequence logic jointly using Conditional Random Fields (CRF) (Lafferty et al., 2001)

5.2.2 DA segmentation Model

The concept we followed in bi-LSTM-CRF sequence labeling is that segmentation is a one-to-one mapping at the character level where each character is annotated as either beginning a segment (B), continues a previous segment (M), ends a segment (E), or is a segment by itself (S). After the labeling is complete we merge the characters and labels together. For example, بيقولوا "byqwlwA" (they say) is labeled as "SBMMEBE", which means that the word is segmented as b+yqwl+wA. The architecture of our segmentation model, shown in Figure 2, is straightforward. At the input layer a look-up table is initialized with randomly uniform sampled embeddings mapping each character in the input to a d-dimensional vector. At the hidden layer, the output from the character embeddings is used as the input to the bi-LSTM layer to obtain fixed-dimensional representations of characters. At the output layer, a CRF is applied on the top of bi-LSTM to jointly decode labels for the whole input characters. Training is performed using stochastic gradient (SGD) descent with momentum 0.9 and batch size 50, optimizing the cross entropy objective function.

Optimization To mitigate overfitting, given the small size of the training data, we employ dropout (Hinton et al., 2012), which prevents co-adaptation of hidden units by randomly setting to zero a proportion of the hidden units during training. We also employ early stopping (Caruana et al., 2000; Graves et al., 2013) by monitoring the models performance on a development set.

6 Experimental Setup and Results

Using the approaches described earlier, we perform several experiments, serving two main objectives. First we want to see how closely related the dialects are and whether we can use one dialect for the augmentation of training data in another dialect. The second objective is to find out whether we can build a one-fits-all model that does not need to know which specific dialect it is dealing with.

In the first set of experiments shown in Table 7, we build segmentation models for each dialect and tested them on all the other dialects. We compare these cross dialect training and testing to training and testing on the same dialect, where we use 5 fold cross validation with 70/10/20 train/dev/test splits. We also use the Farasa MSA segmenter as a baseline. We conduct the experiments at three levels: pure system output (without lookup), with DA lookup, and with DA+MSA lookup. We mean by "lookup" a post-processing add-on step where we feed segmentation solutions in the test files directly from the training data when a match is found. This is based on the assumption that segmentation is a context-free problem and therefore the utilization of observed data can be maximized.

Using both algorithms (SVM and LSTM) the results show a general trend where EGY segmentation yields better results from the LEV model than from the GLF's. The GLF data benefits more from the LEV model than from the EGY one. For the LEV data both GLF and EGY models are equally good. MGR seems relatively distant in that it does not contribute to or benefit from other dialects independently. This shows a trend where dialects favor geographical proximity. In the case with no lookup, LSTM fairs better than SVM when training and testing is done on the same dialect. However, the opposite is true when training on one dialect and testing on another. This may indicate that the SVM-ranker has better cross-dialect generalization than the bi-LSTM-CRF sequence labeler. When lookup is used, SVM yields better results across the board except in three cases, namely when training and testing on Egyptian with DA+MSA lookup, when training with Egyptian and testing on MGR, and when training with GLF and testing on MGR with DA+MSA lookup. Lastly, the best SVM cross-dialect results with lookup consistently beat the Farasa MSA baseline often by several percentage points for every dialect. The same is true for LSTM when training with relatively related dialects (EGY, LEV, and GLF), but the performance decreases when training or testing using MGR.

In the second set of experiments, we wanted to see whether we can train a unified segmenter that would segment all the dialects in our datasets. For the results shown in Table 8, we also used 5-fold cross validation (with the same splits generated earlier) where we trained on the combined training splits from all dialects and tested on all the test splits with no lookup, DA lookup, and MSA+DA lookup. We refer to these models as "joint" models. Using SVM, the combined model drops by 0.3% to 1.3% compared to exclusively using matching dialectal training data. We also conducted another SVM experiment in which we use the joint model in conjunction with a dialect identification oracle to restrict possible affixes only to those that are possible for that dialect (last two row

	Test Set							
Farasa	85.7		82.6		82.9		82.6	
Training	EGY		LEV		GLF		MGR	
	SVM	LSTM	SVM	LSTM	SVM	LSTM	SVM	LSTM
with no lookup								
EGY	91.0	93.8	87.7	87.1	86.5	85.8	81.3	82.5
LEV	85.2	85.5	87.8	91.0	85.5	85.7	83.42	80.0
GLF	85.7	85.0	86.4	86.9	87.7	89.4	82.6	81.6
MGR	85.0	78.6	85.7	78.8	84.5	78.4	84.7	87.1
with DA lookup								
EGY	94.5	94.2	89.2	87.6	87.5	86.5	81.5	82.8
LEV	89.7	85.9	92.9	91.8	89.6	86.3	83.5	80.4
GLF	89.7	85.5	89.2	87.5	92.8	90.8	83.0	82.4
MGR	88.6	78.9	86.9	78.8	87.3	79.0	90.5	88.5
with DA+MSA lookup								
EGY	94.6	**95.0**	90.5	89.2	88.8	88.3	83.5	89.2
LEV	90.1	87.5	**93.3**	93.0	89.7	87.8	84.3	82.4
GLF	90.3	87.3	89.6	88.6	**93.1**	91.9	84.1	84.8
MGR	88.6	81.2	88.1	80.3	88.1	80.7	**91.2**	90.1

Table 7: Cross dialect results.

	Test Set							
Lookup	EGY		LEV		GLF		MGR	
	SVM	LSTM	SVM	LSTM	SVM	LSTM	SVM	LSTM
No lookup	91.4	94.1	89.8	92.4	88.8	91.7	83.82	89.1
DA	94.1	94.8	92.8	93.3	91.8	92.6	89.6	90.7
DA+MSA	94.3	**95.3**	93.0	**93.9**	92.2	**93.1**	90.0	**91.4**
Joint with restricted affixes								
DA	94.5	-	92.8	-	91.9	-	89.7	-
DA+MSA	94.8	-	93.0	-	92.4	-	90.3	-

Table 8: Joint model results.

in Table 8). The results show improvements for all dialects, but aside for EGY, the improvements do not lead to better results than those for single dialect models. Conversely, the bi-LSTM-CRF joint model with DA+MSA lookup beats every other experimental setup that we tested, leading to the best segmentation results for all dialects, without doing dialect identification. This may indicate that bi-LSTM-CRF benefited from cross-dialect data in improving segmentation for individual dialects.

7 Conclusion

This paper presents (to the best of our knowledge) the first comparative study between closely related languages with regards to their segmentation. Arabic dialects diverged from a single origin, yet they maintained pan-dialectal common features which allow them to cross-fertilize.

Our results show that a single joint segmentation model, based on bi-LSTM-CRF, can be developed for a group of dialects and this model yields results that are comparable to, or even superior to, the performance of single dialect-specific models. Our results also show that there is a degree of closeness between dialects that is contingent with the geographical proximity. For example, we statistically show that Gulf is closer to Levantine than to Egyptian, and similarly Levantine is closer to Egyptian than to Gulf. Cross dialect segmentation experiments also show that Maghrebi is equally distant from the other three regional dialects. This sheds some light on the degree of mutual intelligibility between the speakers of Arabic dialects, assuming that the level of success in interdialectal segmentation can be an indicator of how well speakers of the respective dialects can understand each others.

References

Ahmed Abdelali, Kareem Darwish, Nadir Durrani, and Hamdy Mubarak. 2016. Farasa: A fast and furious segmenter for arabic. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*. Association for Computational Linguistics, San Diego, California, pages 11–16.

Mohammed Attia, Pavel Pecina, Antonio Toral, Lamia Tounsi, and Josef van Genabith. 2011. An open-source finite state morphological transducer for modern standard arabic. In *Proceedings of the 9th International Workshop on Finite State Methods and Natural Language Processing*. Association for Computational Linguistics, pages 125–133.

Yoshua Bengio, Patrice Simard, and Paolo Frasconi. 1994. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks* 5(2):157–166.

Fadi Biadsy, Julia Hirschberg, and Nizar Habash. 2009. Spoken arabic dialect identification using phonotactic modeling. In *Proceedings of the EACL 2009 Workshop on Computational Approaches to Semitic Languages*. Association for Computational Linguistics, Stroudsburg, PA, USA, Semitic '09, pages 53–61.

Houda Bouamor, Nizar Habash, and Kemal Oflazer. 2014. A multidialectal parallel corpus of arabic. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*. European Language Resources Association (ELRA), Reykjavik, Iceland.

Tim Buckwalter. 2004. Buckwalter arabic morphological analyzer version 2.0 .

Rich Caruana, Steve Lawrence, and Lee Giles. 2000. Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping. In *NIPS*. pages 402–408.

Kareem Darwish, Hassan Sajjad, and Hamdy Mubarak. 2014. Verifiably effective arabic dialect identification. In *EMNLP*. pages 1465–1468.

Mohamed Eldesouki, Fahim Dalvi, Hassan Sajjad, and Kareem Darwish. 2016. Qcri@ dsl 2016: Spoken arabic dialect identification using textual features. *VarDial 3* page 221.

Alex Graves. 2013. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850* .

Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. 2013. Speech recognition with deep recurrent neural networks. In *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*. IEEE, pages 6645–6649.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks* 18(5):602–610.

Nizar Habash, Mona T Diab, and Owen Rambow. 2012. Conventional orthography for dialectal arabic. In *LREC*. pages 711–718.

Nizar Habash, Ryan Roth, Owen Rambow, Ramy Eskander, and Nadi Tomeh. 2013. Morphological analysis and disambiguation for dialectal arabic. In *Hlt-Naacl*. pages 426–432.

Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580* .

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9(8):1735–1780.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for sequence tagging. *CoRR* abs/1508.01991.

Najib Ismail Jarad. 2014. The grammaticalization of the motion verb "ra" as a prospective aspect marker in syrian arabic. *Al-'Arabiyya* 47:101–118.

Thorsten Joachims. 2006. Training linear svms in linear time. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pages 217–226.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. ICML*.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360* .

Zachary C Lipton, David C Kale, Charles Elkan, and Randall Wetzell. 2015. A critical review of recurrent neural networks for sequence learning. *CoRR* abs/1506.00019.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume*

1: Long Papers). Association for Computational Linguistics, Berlin, Germany, pages 1064–1074. http://www.aclweb.org/anthology/P16-1101.

Mohamed Maamouri, Ann Bies, Seth Kulick, Michael Ciul, Nizar Habash, and Ramy Eskander. 2014. Developing an egyptian arabic treebank: Impact of dialectal morphology on annotation and tool development. In *LREC*. pages 2348–2354.

Emad Mohamed, Behrang Mohit, and Kemal Oflazer. 2012. Annotating and learning morphological segmentation of egyptian colloquial arabic. In *LREC*. pages 873–877.

Will Monroe, Spence Green, and Christopher D Manning. 2014. Word segmentation of informal arabic with domain adaptation. In *ACL (2)*. pages 206–211.

Hamdy Mubarak and Kareem Darwish. 2014. Using twitter to collect a multi-dialectal corpus of arabic. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*. pages 1–7.

Arfath Pasha, Mohamed Al-Badrashiny, Mona Diab, Ahmed El Kholy, Ramy Eskander, Nizar Habash, Manoj Pooleery, Owen Rambow, and Ryan M Roth. 2014. Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of Arabic. *Proc. LREC* .

Maria Persson. 2008. The role of the b-prefix in gulf arabic dialects as a marker of future, intent and/or irrealis 8:26–52.

Younes Samih, Mohammed Attia, Mohamed Eldesouki, Hamdy Mubarak, Ahmed Abdelali, Laura Kallmeyer, and Kareem Darwish. 2017. A neural architecture for dialectal arabic segmentation. *WANLP 2017 (co-located with EACL 2017)* page 46.

Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing* 45(11):2673–2681.

Omar F Zaidan and Chris Callison-Burch. 2014. Arabic dialect identification. *Computational Linguistics* 40(1):171–202.

Rabih Zbib, Erika Malchiodi, Jacob Devlin, David Stallard, Spyros Matsoukas, Richard Schwartz, John Makhoul, Omar F. Zaidan, and Chris Callison-Burch. 2012. Machine translation of arabic dialects. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, Stroudsburg, PA, USA, NAACL HLT '12, pages 49–59.