Word Segmentation of Informal Arabic with Domain Adaptation

Will Monroe, Spence Green, and **Christopher D. Manning**
Computer Science Department, Stanford University
{wmonroe4,spenceg,manning}@stanford.edu

Abstract

Segmentation of clitics has been shown to improve accuracy on a variety of Arabic NLP tasks. However, state-of-the-art Arabic word segmenters are either limited to formal Modern Standard Arabic, performing poorly on Arabic text featuring dialectal vocabulary and grammar, or rely on linguistic knowledge that is hand-tuned for each dialect. We extend an existing MSA segmenter with a simple domain adaptation technique and new features in order to segment informal and dialectal Arabic text. Experiments show that our system outperforms existing systems on newswire, broadcast news and Egyptian dialect, improving segmentation F_1 score on a recently released Egyptian Arabic corpus to 95.1%, compared to 90.8% for another segmenter designed specifically for Egyptian Arabic.

1 Introduction

Segmentation of words, clitics, and affixes is essential for a number of natural language processing (NLP) applications, including machine translation, parsing, and speech recognition (Chang et al., 2008; Tsarfaty, 2006; Kurimo et al., 2006). Segmentation is a common practice in Arabic NLP due to the language's morphological richness. Specifically, clitic separation has been shown to improve performance on Arabic parsing (Green and Manning, 2010) and Arabic-English machine translation (Habash and Sadat, 2006). However, the variety of Arabic dialects presents challenges in Arabic NLP. Dialectal Arabic contains non-standard orthography, vocabulary, morphology, and syntax. Tools that depend on corpora or grammatical properties that only consider formal Modern Standard Arabic (MSA) do not perform well when confronted with these differences. The creation of annotated corpora in dialectal Arabic (Maamouri et al., 2006) has promoted the development of new systems that support dialectal Arabic, but these systems tend to be tailored to specific dialects and require separate efforts for Egyptian Arabic, Levantine Arabic, Maghrebi Arabic, etc.

We present a single clitic segmentation model that is accurate on both MSA and informal Arabic. The model is an extension of the character-level conditional random field (CRF) model of Green and DeNero (2012). Our work goes beyond theirs in three aspects. First, we handle two Arabic orthographic normalization rules that commonly require rewriting of tokens after segmentation. Second, we add new features that improve segmentation accuracy. Third, we show that dialectal data can be handled in the framework of *domain adaptation*. Specifically, we show that even simple feature space augmentation (Daumé, 2007) yields significant improvements in task accuracy.

We compare our work to the original Green and DeNero model and two other Arabic segmentation systems: the MADA+TOKAN toolkit v. 3.1 (Habash et al., 2009) and its Egyptian dialect variant, MADA-ARZ v. 0.4 (Habash et al., 2013). We demonstrate that our system achieves better performance across the board, beating all three systems on MSA newswire, informal broadcast news, and Egyptian dialect. Our segmenter achieves a 95.1% F_1 segmentation score evaluated against a gold standard on Egyptian dialect data, compared to 90.8% for MADA-ARZ and 92.9% for Green and DeNero. In addition, our model decodes input an order of magnitude faster than either version of MADA. Like the Green and DeNero system, but unlike MADA and MADA-ARZ, our system does not rely on a morphological analyzer, and can be applied directly to any dialect for which segmented training data is available. The source code is available in the latest public release of the Stanford Word Segmenter (`http://nlp.stanford.edu/software/segmenter.shtml`).

Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Short Papers), pages 206–211,
Baltimore, Maryland, USA, June 23-25 2014.

2 Arabic Word Segmentation Model

A CRF model (Lafferty et al., 2001) defines a distribution $p(\mathbf{Y}|\mathbf{X};\theta)$, where $\mathbf{X} = \{x_1, \ldots, x_N\}$ is the observed input sequence and $\mathbf{Y} = \{y_1, \ldots, y_N\}$ is the sequence of labels we seek to predict. Green and DeNero use a linear-chain model with $\mathbf{X}$ as the sequence of input *characters*, and $\mathbf{Y}^*$ chosen according to the decision rule

$$\mathbf{Y}^* = \arg\max_{\mathbf{Y}} \sum_{i=1}^{N} \theta^\top \phi(\mathbf{X}, y_i, \ldots, y_{i-3}, i) \; .$$

where ϕ is the feature map defined in Section 2.1. Their model classifies each y_i as one of I (continuation of a segment), O (whitespace outside any segment), B (beginning of a segment), or F (pre-grouped foreign characters).

Our segmenter expands this label space in order to handle two Arabic-specific orthographic rules. In our model, each y_i can take on one of the six values $\{\text{I}, \text{O}, \text{B}, \text{F}, \text{RewAl}, \text{RewTa}\}$:

- RewAl indicates that the current character, which is always the Arabic letter ل *l*, starts a new segment and should additionally be transformed into the definite article الـ *al-* when segmented. This type of transformation occurs after the prefix لـ *li-* "to".
- RewTa indicates that the current character, which is always the Arabic letter ت *t*, is a continuation but should be transformed into the letter ة *h* when segmented. Arabic orthography rules restrict the occurrence of ة *h* to the word-final position, writing it instead as ت *t* whenever it is followed by a suffix.

2.1 Features

The model of Green and DeNero is a third-order (i.e., 4-gram) Markov CRF, employing the following indicator features:

- a five-character window around the current character: for each $-2 \le \delta \le 2$ and $1 \le i \le N$, the triple $(x_{i+\delta}, \delta, y_i)$
- n-grams consisting of the current character and up to three preceding characters: for each $2 \le n \le 4$ and $n \le i \le N$, the character-sequence/label-sequence pair $(x_{i-n+1} \ldots x_i, y_{i-n+1} \ldots y_i)$
- whether the current character is punctuation
- whether the current character is a digit
- the Unicode block of the current character
- the Unicode character class of the current character

In addition to these, we include two other types of features motivated by specific errors the original system made on Egyptian dialect development data:

- Word length and position within a word: for each $1 \le i \le N$, the pairs (ℓ, y_i), (a, y_i), and (b, y_i), where ℓ, a, and b are the total length of the word containing x_i, the number of characters after x_i in the word, and the number of characters before x_i in the word, respectively. Some incorrect segmentations produced by the original system could be ruled out with the knowledge of these statistics.
- First and last two characters of the current word, separately influencing the first two labels and the last two labels: for each word consisting of characters $x_s \ldots x_t$, the tuples $(x_s x_{s+1},\ x_{t-1} x_t,\ y_s y_{s+1},$ "begin") and $(x_s x_{s+1},\ x_{t-1} x_t,\ y_{t-1} y_t,$ "end"). This set of features addresses a particular dialectal Arabic construction, the negation ما *mā-* + [verb] + ش *-sh*, which requires a matching prefix and suffix to be segmented simultaneously. This feature set also allows the model to take into account other interactions between the beginning and end of a word, particularly those involving the definite article الـ *al-*.

A notable property of this feature set is that it remains highly dialect-agnostic, even though our additional features were chosen in response to errors made on text in Egyptian dialect. In particular, it does not depend on the existence of a dialect-specific lexicon or morphological analyzer. As a result, we expect this model to perform similarly well when applied to other Arabic dialects.

2.2 Domain adaptation

In this work, we train our model to segment Arabic text drawn from three domains: newswire, which consists of formal text in MSA; broadcast news, which contains scripted, formal MSA as well as extemporaneous dialogue in a mix of MSA and dialect; and discussion forum posts written primarily in Egyptian dialect.

Model	Training Data	F_1 (%) ATB	BN	ARZ	TEDEval (%) ATB	BN	ARZ
GD	ATB	97.60	94.87	79.92	98.22	96.81	87.30
GD	+BN+ARZ	97.28	96.37	92.90	98.05	97.45	95.01
+Rew	ATB	97.55	94.95	79.95	98.72	97.45	87.54
+Rew	+BN	97.58	96.60	82.94	98.75	98.18	89.43
+Rew	+BN+ARZ	97.30	96.09	92.64	98.59	97.91	95.03
+Rew+DA	+BN+ARZ	97.71	96.57	93.87	98.79	98.14	95.86
+Rew+DA+Feat	+BN+ARZ	**98.36**	**97.35**	**95.06**	**99.14**	**98.57**	**96.67**

Table 1: Development set results. **GD** is the model of Green and DeNero (2012). **Rew** is support for orthographic rewrites with the REWAL and REWTA labels. The fifth row shows the strongest baseline, which is the GD+Rew model trained on the concatenated training sets from all three treebanks. **DA** is domain adaptation via feature space augmentation. **Feat** adds the additional feature templates described in section 2.1. **ATB** is the newswire ATB; **BN** is the Broadcast News treebank; **ARZ** is the Egyptian treebank. Best results (**bold**) are statistically significant ($p < 0.001$) relative to the strongest baseline.

The approach to domain adaptation we use is that of *feature space augmentation* (Daumé, 2007). Each indicator feature from the model described in Section 2.1 is replaced by $N + 1$ features in the augmented model, where N is the number of domains from which the data is drawn (here, $N = 3$). These $N + 1$ features consist of the original feature and N "domain-specific" features, one for each of the N domains, each of which is active only when both the original feature is present and the current text comes from its assigned domain.

3 Experiments

We train and evaluate on three corpora: parts 1–3 of the newswire Arabic Treebank (ATB),[1] the Broadcast News Arabic Treebank (BN),[2] and parts 1–8 of the BOLT Phase 1 Egyptian Arabic Treebank (ARZ).[3] These correspond respectively to the domains in section 2.2. We target the segmentation scheme used by these corpora (leaving morphological affixes and the definite article attached). For the ATB, we use the same split as Chiang et al. (2006). For each of the other two corpora, we split the data into 80% training, 10% development, and 10% test in chronological order by document.[4] We train the Green and DeNero model and our improvements using L-BFGS with L_2 regularization.

[1] LDC2010T13, LDC2011T09, LDC2010T08

[2] LDC2012T07

[3] LDC2012E{93,98,89,99,107,125}, LDC2013E{12,21}

[4] These splits are publicly available at `http://nlp.stanford.edu/software/parser-arabic-data-splits.shtml`.

3.1 Evaluation metrics

We use two evaluation metrics in our experiments. The first is an F_1 precision-recall measure, ignoring orthographic rewrites. F_1 scores provide a more informative assessment of performance than word-level or character-level accuracy scores, as over 80% of tokens in the development sets consist of only one segment, with an average of one segmentation every 4.7 tokens (or one every 20.4 characters).

The second metric we use is the TEDEval metric (Tsarfaty et al., 2012). TEDEval was developed to evaluate joint segmentation and parsing[5] in Hebrew, which requires a greater variety of orthographic rewrites than those possible in Arabic. Its edit distance-based scoring algorithm is robust enough to handle the rewrites produced by both MADA and our segmenter.

We measure the statistical significance of differences in these metrics with an approximate randomization test (Yeh, 2000; Padó, 2006), with $R = 10{,}000$ samples.

3.2 Results

Table 1 contains results on the development set for the model of Green and DeNero and our improvements. Using domain adaptation alone helps performance on two of the three datasets (with a statistically insignificant decrease on broadcast news), and that our additional features further improve

[5] In order to evaluate segmentation in isolation, we convert each segmented sentence from both the model output and the gold standard to a flat tree with all segments descending directly from the root.

	F_1 (%)			TEDEval (%)		
	ATB	BN	ARZ	ATB	BN	ARZ
MADA	97.36	94.54	78.35	97.62	96.96	86.78
MADA-ARZ	92.83	91.89	90.76	91.26	91.10	90.39
GD+Rew+DA+Feat	**98.30**	**97.17**	**95.13**	**99.10**	**98.42**	**96.75**

Table 2: Test set results. Our final model (last row) is trained on all available data (ATB+BN+ARZ). Best results (**bold**) are statistically significant ($p < 0.001$) relative to each MADA version.

	ATB	BN	ARZ
MADA	705.6 ± 5.1	472.0 ± 0.8	767.8 ± 1.9
MADA-ARZ	784.7 ± 1.6	492.1 ± 4.2	779.0 ± 2.7
GD+Rew+DA+Feat	**90.0** ± 1.0	**59.5** ± 0.3	**72.7** ± 0.2

Table 3: Wallclock time (in seconds) for MADA, MADA-ARZ, and our model for decoding each of the three development datasets. Means and standard deviations were computed for 10 independent runs. MADA and MADA-ARZ are single-threaded. Our segmenter supports multithreaded execution, but the times reported here are for single-threaded runs.

segmentation on all datasets. Table 2 shows the segmentation scores our model achieves when evaluated on the three test sets, as well as the results for MADA and MADA-ARZ. Our segmenter achieves higher scores than MADA and MADA-ARZ on all datasets under both evaluation metrics. In addition, our segmenter is faster than MADA. Table 3 compares the running times of the three systems. Our segmenter achieves a 7x or more speedup over MADA and MADA-ARZ on all datasets.

4 Error Analysis

We sampled 100 errors randomly from all errors made by our final model (trained on all three datasets with domain adaptation and additional features) on the ARZ development set; see Table 4. These errors fall into three general categories:

- typographical errors and annotation inconsistencies in the gold data;
- errors that can be fixed with a fuller analysis of just the problematic token, and therefore represent a deficiency in the feature set; and
- errors that would require additional context or sophisticated semantic awareness to fix.

4.1 Typographical errors and annotation inconsistencies

Of the 100 errors we sampled, 33 are due to typographical errors or inconsistencies in the gold data. We classify 7 as typos and 26 as annotation inconsistencies, although the distinction between the two is murky: typos are intentionally preserved in the treebank data, but segmentation of typos varies depending on how well they can be reconciled with standard Arabic orthography. Four of the seven typos are the result of a missing space, such as:

- يسهربالّليالي *yashar-bi-'l-layālī* "staysawakeatnight" (يسهر *yashar* + بـ *bi-* + اللّيالي *al-layālī*)
- عملتناأن *ʿamilatnā-ʾan* "madeus" (عملت *ʿamilat* + نا *-nā* + أن *ʾan*)

The first example is segmented in the Egyptian treebank but is left unsegmented by our system; the second is left as a single token in the treebank but is split into the above three segments by our system.

Of the annotation inconsistencies that do not involve typographical errors, a handful are segmentation mistakes; however, in the majority of these cases, the annotator chose not to segment a word for justifiable but arbitrary reasons. In particular, a few colloquial "filler" expressions are sometimes not segmented, despite being compound Arabic words that are segmented elsewhere in the data. These include ربّنا *rabbinā* "[our] Lord" (oath); عندما *ʿindamā* "when"/"while"; and خلّيك *khallīk* "keep"/"stay". Also, tokens containing foreign words are sometimes not segmented, despite carrying Arabic affixes. An example of this is ومستر

Category	# of errors
Abnormal gold data	**33**
Typographical error	*7*
Annotation inconsistency	*26*
Need full-token features	**36**
Need more context	**31**
ولا *wlā*	*5*
ـنا *-nā*: verb/pron	*7*
ـي *-y*: *nisba*/pron	*4*
other	*15*

Table 4: Counts of error categories (out of 100 randomly sampled ARZ development set errors).

wamistur "and *Mister* [English]", which could be segmented as و *wa-* + مستر *mistur*.

4.2 Features too local

In 36 of the 100 sampled errors, we conjecture that the presence of the error indicates a shortcoming of the feature set, resulting in segmentations that make sense locally but are not plausible given the full token. Two examples of these are:

- وفطريقة *wafiṭarīqah* "and in the way" segmented as و *wa-* + فطريقة *fiṭarīqah* (correct analysis is و *wa-* + فـ *fi-* + طريقة *ṭarīqah*). فطر *fṭr* "break"/"breakfast" is a common Arabic root, but the presence of ق *q* should indicate that فطر *fṭr* is not the root in this case.

- ولايهمّهم *walāyuhimmhum* "and it's not important to them" segmented as و *wa-* + لـ *li-* + ـايهمّ *-ayuhimm* + ـهم *-hum* (correct analysis is و *wa-* + لا *lā* + يهمّ *yuhimm* + ـهم *-hum*). The 4-character window لايه *lāyh* occurs commonly with a segment boundary after the ل *l*, but the segment ـايهمّ *-ayuhimm* is not a well-formed Arabic word.

4.3 Context-sensitive segmentations and multiple word senses

In the remaining 31 of 100 errors, external context is needed. In many of these, it is not clear how to address the error without sophisticated semantic reasoning about the surrounding sentence.

One token accounts for five of these errors: ولا *wlā*, which in Egyptian dialect can be analyzed as و *wa-* + لا *lā* "and [do/does] not" or as ولّا *wallā* "or". In a few cases, either is syntactically correct, and the meaning must be inferred from context.

Two other ambiguities are a frequent cause of error and seem to require sophisticated disambiguation. The first is ـنا *-nā*, which is both a first person plural object pronoun and a first person plural past tense ending. The former is segmented, while the latter is not. An example of this is the pair علمنا *ʿilmunā* "our knowledge" (علم *ʿilmu* + ـنا *-nā*) versus علمنا *ʿalimnā* "we knew" (one segment). The other is ـي *-y*, which is both a first person singular possessive pronoun and the *nisba* adjective ending (which turns a noun into an adjective meaning "of or related to"); only the former is segmented. One example of this distinction that appeared in the development set is the pair موضوعي *mawḍūʿī* "my topic" (موضوع *mawḍūʿ* + ـي *-y*) versus موضوعيّ *mawḍūʿīy* "topical", "objective".

5 Conclusion

In this paper we demonstrate substantial gains on Arabic clitic segmentation for both formal and dialectal text using a single model with dialect-independent features and a simple domain adaptation strategy. We present a new Arabic segmenter which performs better than tools employing sophisticated linguistic analysis, while also giving impressive speed improvements. We evaluated our segmenter on broadcast news and Egyptian Arabic due to the current availability of annotated data in these domains. However, as data for other Arabic dialects and genres becomes available, we expect that the model's simplicity and the domain adaptation method we use will allow the system to be applied to these dialects with minimal effort and without a loss of performance in the original domains.

Acknowledgments

We thank the three anonymous reviewers, and Reut Tsarfaty for valuable correspondence regarding TEDEval. The second author is supported by a National Science Foundation Graduate Research Fellowship. This work was supported by the Defense Advanced Research Projects Agency (DARPA) Broad Operational Language Translation (BOLT) program through IBM. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the view of DARPA or the US government.

References

Pi-Chuan Chang, Michel Galley, and Christopher D. Manning. 2008. Optimizing Chinese word segmentation for machine translation performance. In *WMT*.

David Chiang, Mona T. Diab, Nizar Habash, Owen Rambow, and Safiullah Shareef. 2006. Parsing Arabic dialects. In *EACL*.

Hal Daumé, III. 2007. Frustratingly easy domain adaptation. In *ACL*.

Spence Green and John DeNero. 2012. A class-based agreement model for generating accurately inflected translations. In *ACL*.

Spence Green and Christopher D. Manning. 2010. Better Arabic parsing: Baselines, evaluations, and analysis. In *COLING*.

Nizar Habash and Fatiha Sadat. 2006. Arabic preprocessing schemes for statistical machine translation. In *NAACL, Short Papers*.

Nizar Habash, Owen Rambow, and Ryan Roth. 2009. MADA+TOKAN: A toolkit for Arabic tokenization, diacritization, morphological disambiguation, POS tagging, stemming and lemmatization. In *MEDAR*.

Nizar Habash, Ryan Roth, Owen Rambow, Ramy Eskander, and Nadi Tomeh. 2013. Morphological analysis and disambiguation for dialectal Arabic. In *HLT-NAACL*.

Mikko Kurimo, Antti Puurula, Ebru Arisoy, Vesa Siivola, Teemu Hirsimäki, Janne Pylkkönen, Tanel Alumäe, and Murat Saraclar. 2006. Unlimited vocabulary speech recognition for agglutinative languages. In *HLT-NAACL*.

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

Mohamed Maamouri, Ann Bies, Tim Buckwalter, Mona Diab, Nizar Habash, Owen Rambow, and Dalila Tabessi. 2006. Developing and using a pilot dialectal Arabic treebank. In *LREC*.

Sebastian Padó, 2006. *User's guide to `sigf`: Significance testing by approximate randomisation*. `http://www.nlpado.de/~sebastian/software/sigf.shtml`.

Reut Tsarfaty, Joakim Nivre, and Evelina Andersson. 2012. Joint evaluation of morphological segmentation and syntactic parsing. In *ACL, Short Papers*.

Reut Tsarfaty. 2006. Integrated morphological and syntactic disambiguation for Modern Hebrew. In *COLING-ACL*.

Alexander Yeh. 2000. More accurate tests for the statistical significance of result differences. In *COLING*.