Minimally Supervised Morphological Segmentation with Applications to Machine Translation

Jason Riesa and **David Yarowsky**
Department of Computer Science
Johns Hopkins University
Baltimore, MD 21218
{riesa, yarowsky}@jhu.edu

Abstract

Inflected languages in a low-resource setting present a data sparsity problem for statistical machine translation. In this paper, we present a minimally supervised algorithm for morpheme segmentation on Arabic dialects which reduces unknown words at translation time by over 50%, total vocabulary size by over 40%, and yields a significant increase in BLEU score over a previous state-of-the-art phrase-based statistical MT system.

1 Introduction

A significant problem for statistical machine translation in a low-resource setting is data sparsity caused by highly inflected languages. Languages such as Arabic and Turkish, for example, are rich with complex morphology. For statistical machine translation, this means much more parallel data is required in order to accurately learn translations for the increased number of unique words and phrases that result.

In this paper, we focus on the problem of prefix and suffix morpheme segmentation with the ultimate goal of increasing the accuracy of statistical machine translation systems by reducing the amount of unknown words at translation time as well as reducing total vocabulary size. Table 1 shows that it is nontrivial to make a correct analysis by inspection of the word surface form alone. A naïve rule-based scheme employing a stemmed word list may yield adequate results, but only if the word list is complete and covers all unique word types found in a corpus. Not only are these resources difficult to obtain for low-resource languages, such methods are not robust to gaps in the lexicon and fail to indicate a context-based preference when multiple analyses are present.

Possible Affix	Inflected Analysis	Uninflected Analysis	Correct Gloss
بـ [b]	**b+ syArp**	bsyArp	*by car*
بـ [b]	b+ TAryAt	**bTAryAt**	*batteries*
الـ [Al]	**Al+ mATwrAt**	AlmATwrAt	*the motors*
الـ [Al]	Al+ ly	**Ally**	*that/which*
ـي [y]	**Ax +y**	Axy	*my brother*
ـي [y]	mksyk +y	**mksyky**	*Mexican*

Table 1. Examples of correct and incorrect Iraqi Arabic morpheme segmentation analyses for three different affixes: بـ [b], الـ [Al], and ـي [y]. The correct analysis in each case is shown in bold.

With the aid of a small lexicon annotated for morphological segmentation and part-of-speech, we employ a supervised trie-based segmentation model trained on relatively little data. This model for morpheme segmentation is then evaluated on both Iraqi and Levantine Arabic – two dialectal variants of Modern Standard Arabic (MSA). Few linguistic resources and little parallel or monolingual data are currently available for these two dialects. We

Proceedings of the 7th Conference of the Association for Machine Translation in the Americas, pages 185-192,
Cambridge, August 2006.

Corpus Statistic	**English**	**Iraqi**	**Levantine**
Utterances	36,895	36,895	43,471
Running words	438,911	305,889	247,741
Words per utterance	11.9	8.3	5.7
Unique words	8,776	29,238	26,147

Table 2. Summary statistics for the Levantine Arabic and the English/Iraqi Arabic speech corpora.

show that with little modification for each dialect, and minimal data for training, our model yields improved segmentation accuracy over a standard rule-based lexicon approach described in (Riesa et al., 2006). In addition, the trained model is applied to an English-Iraqi Arabic parallel corpus of 36,895 utterances and yields a significant improvement in BLEU score (Papineni et al., 2002) over a baseline system without use of morphological segmentation.

2 Related Work

Lee et al. (2003) present an algorithm for morpheme segmentation seeded by a 110,000-word manually segmented corpus. Subsequently, Lee (2004) shows that, in conjunction with manual deletion of the Arabic article ال [Al] in some instances, his algorithm yields improved translation accuracy over a baseline phrase-based SMT system. Buckwalter (2004) presents a lexicon-based morphological analyzer for MSA. A publicly available tool by Diab (2004) trained on a large amount MSA newswire using a Support Vector Machine (SVM) based approach, provides part-of-speech tagging and morpheme segmentation as an intermediate step. Habash and Rambow (2005) present an SVM-based classifier approach to morphological analysis for MSA trained on the Penn Arabic Treebank (Maamouri et al., 2004). Habash et al. (2005) describe a work in progress implementing a morphological analysis tool involving the modeling of Arabic morphological and phonemic phenomena to provide morphological analysis taking into account the nonstandard orthography found in transcribed data for many spoken dialects of Arabic.

3 Dialectal Speech Corpora

We perform our experiments and evaluate our model on two low-resource dialects of Arabic: Iraqi and Levantine Arabic. The Iraqi Arabic corpus is from the Defense Advanced Research Projects Agency (DARPA) Transtac program, derived from 40 hours of recorded and transcribed audio. The Levantine Arabic corpus is derived from the Levantine Arabic Conversational Telephone Speech Collection, part of the DARPA EARS (Effective, Affordable, Reusable Speech-to-Text) program – also a compilation of 40 hours of audio transcription. Because both are corpora wholly consisting of transcribed speech, disfluencies and metalinguistic tags have been removed.

In addition to the monolingual data above, the Iraqi Arabic corpus has also been translated into English. We use this English/Iraqi Arabic parallel corpus to evaluate the usefulness, in a sparse-data setting, of applying the morpheme segmentation method discussed below to a parallel corpus prior to end-to-end machine translation. Table 2 shows summary statistics for the English/Iraqi Arabic parallel corpus and the Levantine Arabic corpus.

Compiling Training and Test Sets for Morpheme Segmentation

In addition to the above unannotated corpora described above, we make use of two small monolingual lexicons for each dialect. Each lexicon is annotated for part-of-speech and morphology, and each contains only a subset of the total unique word types found in the respective corpora. The Iraqi Arabic lexicon has coverage of 26% of all types and 80% of all tokens found in the Iraqi corpus; the Levantine Arabic lexicon has coverage of 20% of all types and 56% of all tokens found in the Levantine corpus.

For each affix a to be segmented, two lists of words are compiled from the lexicon.

1. The *inflected* exemplar list: A list of words beginning or ending with the string a and whose

morphological annotation indicates a is indeed an affix, and not part of the base word.

2. The *uninflected* exemplar list: A list of words beginning or ending with the string a and whose morphological annotation indicates that a is not an affix, and is part of the base word.

Note that if a is a prefix, only the word-initial position a word is inspected for the string a. Analogously, if a is a suffix, only the word-final position of the word is inspected.

Next, for each word w that does not appear in both lists, we gather its adjacent local contexts. We use the two words immediately preceding and following w. These contexts are easily extracted from an enumeration of all corpus trigrams, and form two more lists: (1) the *inflected* context list, and (2) the *uninflected* context list.

Approximately 20% of words from each exemplar list are held out for the development and test sets – 10% for each set. The development set will be used for training the parameters of the model; the test set is used in the final evaluation.

4 Trie-based Model for Morphological Segmentation

We assume an input Arabic word takes the form $r_1 r_2 ... r_n w s_1 s_2 ... s_m$, where w is inflected by n prefixes and m suffixes, and use supervised trie-based classifier models trained on relatively small amounts of data to perform morpheme segmentation. Table 3 shows the affixes we consider in this work. Each is an inflectional morpheme, and each generally aligns naturally to an English word when segmented. One model is built and trained for each affix segmentation problem considered in isolation.

Each classifier, given an input word w, makes a binary decision regarding whether or not the input word is inflected with a certain affix a. To make this decision, the model consults (1) the prior probability of word w being inflected with affix a (computed offline), and (2) four character tries:

1. A *prefix trie*, in which all words in the inflected and uninflected exemplar lists for each affix are pushed down the trie in left-to-right order, starting with the word-initial character and ending with the word-final character. This follows since we are using the Buckwalter transliteration scheme for Arabic to convert the original orthography to the Roman alphabet. Thus, in using Arabic orthography, one would push words down this trie in right-to-left order.

2. A *suffix trie*, in which all words in the inflected and uninflected exemplar lists for each affix are pushed down the trie in the opposite order of the prefix trie.

3. A *pre-context trie*, initialized by pushing all the pre-context words from the pre-context list down the trie in right-to-left-order

4. A *post-context trie*, initialized by pushing all the post-context words from the post-context list down the trie in left-to-right order.

Prefix	Gloss	Suffix	Gloss
الـ [Al]	the+	ـي [y]	+1-sg-pron
و [w]	and+	ـني [ny]	+1-sg-pron (verbal)
لـ [l]	for+	ـك [k]	+2-sg-pron
بـ [b]	to/in+	ـه [h]	+3-sg-masc-pron
فـ [f]	so/then+	ـها [hA]	+3-sg-fem-pron
شـ [$]	what+	ـنا [nA]	+1-pl-pron
ما [mA]	neg+	ـكم [km]	+2-pl-masc-pron
مو [mw]	neg+	ـكن [kn]	+2-pl-fem-pron
لا [lA]	neg+	ـهم [hm]	+3-pl-masc-pron
للـ [ll]	for+the+	ـهن [hn]	+3-pl-fem-pron
هـ [h]	this+		
عـ [E]	on+		

Table 3. Arabic affixes considered by our morpheme segmentation method. Note that Levantine Arabic does not make use of feminine plural pronouns.

4.1 The Trie

Figure 1 shows the components of a trie in this framework. Each node in the trie corresponds to a character seen in training and holds a frequency distribution which counts the number of times this character was seen in an inflected word, and the number of times seen in an uninflected word. The ϵ marker terminates all strings, and since this trie is built for the **y** classifier, there is only one child of the root

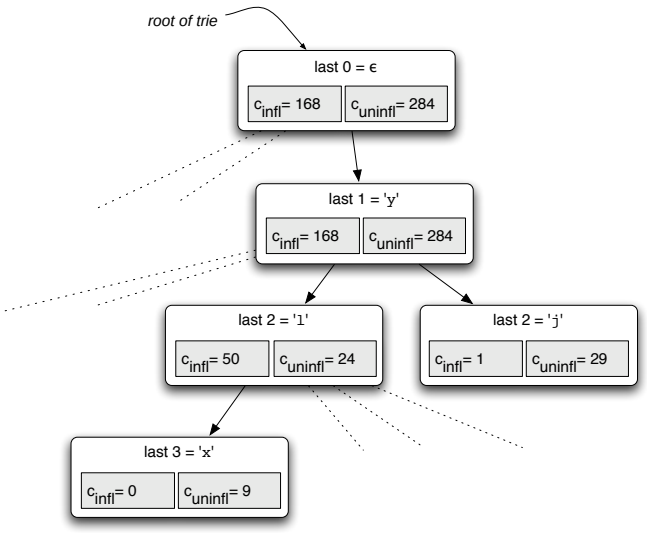


Figure 1. Diagram of suffix trie in the classifier for suffix morpheme **y**. Traversing farther down this trie is analogous to moving from word-final position to word-initial position in right-to-left order. Adapted from (Wicentowski, 2002).

node. At subsequent levels of the trie, the frequency counts become more restricted as the trie becomes more confident as to the class (*inflected* or *uninflected*) of a given word. For each character pushed down the trie and ending up in some node n, the appropriate frequency counter for n is incremented depending on the class of the word.

In order to evaluate the probability that a given word $\mathbf{w} = w_1 w_2 ... w_n$ is inflected with affix a, each of the four tries from classifier a first returns a probability of its own. For a = suffix **y**, we call these $p^{\mathbf{y}}_{px}$, $p^{\mathbf{y}}_{sx}$, $p^{\mathbf{y}}_{lc}$, $p^{\mathbf{y}}_{rc}$. Then the probabilities representing the degree to which **w** is not inflected by affix a are $1 - p^{\mathbf{y}}_{px}$, $1 - p^{\mathbf{y}}_{sx}$, $1 - p^{\mathbf{y}}_{lc}$, $1 - p^{\mathbf{y}}_{rc}$. Consider computing, for example, $p^{\mathbf{y}}_{px}(\mathbf{w})$, the probability that the prefix trie says word **w** is inflected with affix **y**:

First, we compute the smoothed prefix trie frequencies for **w** for each class as in Cucerzan and Yarowsky (1999):

$$\hat{f}^{INFL}_{px} = \beta f_{px}(class_{INFL}|w_1) + \sum_{i=2}^{m} \alpha^{m-i} f(class_{INFL}|w_1 w_2 ... w_i) \quad (1)$$

... داسمع ناس ببيتك وارید اعرف ...

... I'm hearing people in your house and I want to know ...

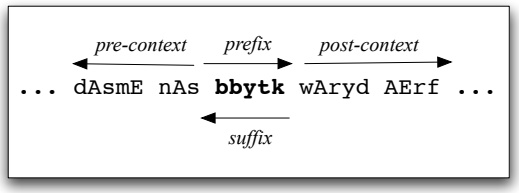


Figure 2. This diagram shows the directions in which words and their local contexts are pushed down the four tries in the model.

$$\hat{f}^{UNINFL}_{px} = \beta f_{px}(class_{UNINFL}|w_1) + \sum_{i=2}^{m} \alpha^{m-i} f(class_{UNINFL}|w_1 w_2 ... w_i) \quad (2)$$

where $\alpha, \beta \in (0, 1)$. Note β is often a small number since generally the first character in a word pushed down the trie is less discriminative than subsequent characters down the path of the same trie. Note also, m is the length of the path through the trie that matches the most characters in $\mathbf{w} = w_1 w_2 ... w_n$.

Finally, we normalize to get $p^{\mathbf{y}}_{px}(\mathbf{w})$:

$$p^{\mathbf{y}}_{px}(\mathbf{w}) = \frac{\hat{f}^{INFL}_{px}}{\hat{f}^{INFL}_{px} + \hat{f}^{UNINFL}_{px}} \quad (3)$$

After computing $p^{\mathbf{y}}_{px}$, $p^{\mathbf{y}}_{sx}$, $p^{\mathbf{y}}_{lc}$, $p^{\mathbf{y}}_{rc}$, the model returns a final consensus probability $\hat{p}^{\mathbf{y}}$ computed as in Equation 4:

$$\hat{p}^{\mathbf{y}} = \eta \cdot (\gamma_1 \cdot p^{\mathbf{y}}_{px} + \gamma_2 \cdot p^{\mathbf{y}}_{sx} + \gamma_3 \cdot p^{\mathbf{y}}_{lc} + \gamma_4 \cdot p^{\mathbf{y}}_{rc}) + (1 - \eta) \cdot p^{\mathbf{y}}_{prior} \quad (4)$$

where

$$\gamma_1, \gamma_2, \gamma_3, \eta \in [0, 1]$$
$$\gamma_4 = 1 - \gamma_1 - \gamma_2 - \gamma_3$$
$$\sum_{j=1}^{4} \gamma_j = 1.$$

Thus, if $\hat{p}^{\mathbf{y}} > \tau$, where τ is some threshold, then we are confident enough to segment affix **y** from word **w**. In these experiments we set $\tau = 0.5$, which is reasonable for a binary classifier. Threshold τ can be increased for more conservative segmentation results.

4.2 Parameter Training

Each trie in an affix classifier contains the two parameters α and β for computing the smoothed trie frequency. In addition, each affix classifier has four parameters that essentially serve as weights in computing the final consensus probability. The γ_j's weight the trie probabilities, and η indicates how much weight to give the prior.

These parameters are trained with an exhaustive search of the parameter space [0,1], discretized into intervals of 0.01. For each classifier, the parameter values are chosen to maximize the accuracy of that classifier on the held-out development set. In order to break ties among parameter value assignments that yield ties in classification accuracy, the assignment that minimizes the cross-entropy of the classifier's hypotheses is chosen.

4.3 Cascading Classifiers for Segmentation of Multiple Morphemes

In Arabic, and many other languages, words may carry more than one affix. In this section we describe cascading the decisions of the above classifiers to provide morpheme segmentation for many affixes.

Figure 3 shows the end-to-end state diagram for this process. Consider the Arabic word `wbbytk`, meaning *and to your house*. Using hyphens to denote places of morpheme affixation, `wbbytk` has true segmentation: `w- b- byt -k` with two prefixes and one suffix.

We start by pushing `wbbytk` through the system and checking to see if there is any prefix in the word-initial position that can be segmented. Since there is a match for prefix `w` here, we send the word through the `w`-classifier to get $\hat{p}^{\mathtt{w}}(\text{“}\mathtt{wbbytk}'') = p_1$. If $p_1 > 0.5$ then we segment `w` to get `w- bbytk`, move on and attempt to segment another prefix unless there are no more matches. If $\hat{p}^{\mathtt{w}}(\text{“}\mathtt{wbbytk}'') = p_1 \leqslant 0.5$, we decide that `w` is indeed part of the base or stemmed word, stop looking to segment any more prefixes, and move on to check for possible suffixes. If there is more than one prefix match in the word-initial position, the process continues greedily by segmenting the prefix whose classifier returns the highest probability of segmentation. As Figure 3 shows, after segmenting all possible prefixes, ideally we arrive at the correct final analysis of `w- b- byt -k` with probability $p_1 \cdot p_2 \cdot (1 - p_3) \cdot p_4$.

We then push `wbbytk` through the system in the opposite direction, starting with decisions from the suffix classifiers and ending with decisions from the prefix classifiers. This direction also returns a probability for its final segmentation, and the analysis with the maximum of these forward and backward probabilities is adopted.

5 Evaluation

For each affix in Table 3, we evaluate the accuracy of the trie-based model using the held out test set described in Section 1. We compare the performance of the trie model to a simple rule-based approach baseline. In addition, we evaluate the performance of a hybrid model which first uses the rule-based scheme lexicon matching for segmentation, but backs off to the trie model for a segmentation decision when no analysis can be made due to gaps in the lexicon.

The rule-based scheme employs a lexicon of

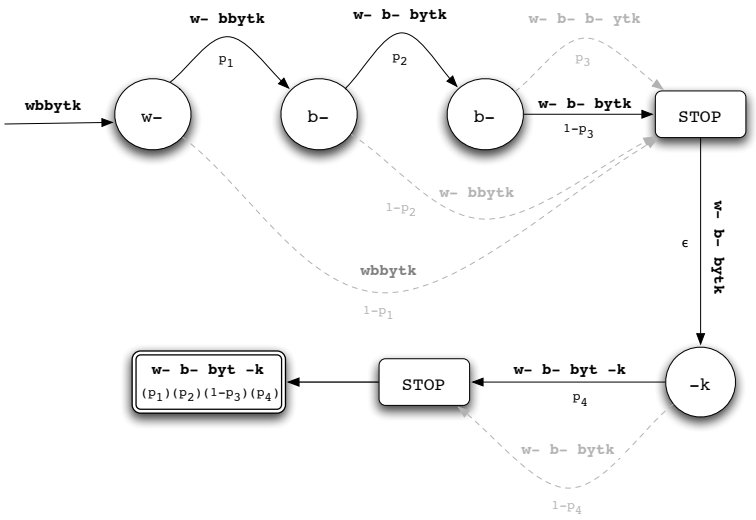


Figure 3. Diagram of end-to-end cascading classifier process. Solid arrows denote correct decisions by each classifier, while dotted arrows denote incorrect paths. This figure depicts the Arabic word `wbbytk` being sent through the end-to-end system in the forward direction, with classifier decisions regarding prefixes being made before those regarding suffixes. Each decision is made with some probability p_j.

stemmed words, compiled from the monolingual dialectal corpora and a large MSA dictionary by inserting words into the wordlist that do not appear to carry any known affix. The performance of this rule-based lexicon approach is shown in Figures 4(a) and 4(b) for Levantine Arabic and Iraqi Arabic, respectively. Accuracy measurements are taken at periodic intervals as we increase the size of the stemmed lexicon. These figures show the segmentation accuracy of the three models averaged over all morphemes. Recall from Section 1 that the test set size for some affix a is 10% of the all words in the lexicon that carry that affix. Thus, in order to give higher importance to the segmentation performance of more common affixes, we perform a weighted average of the accuracies over all affixes using the size of each test set as the weights. Finally, in addition to the trie, hybrid and lexicon models, the performance of using solely the prior probability for each affix is shown for comparison.

5.1 Levantine Arabic

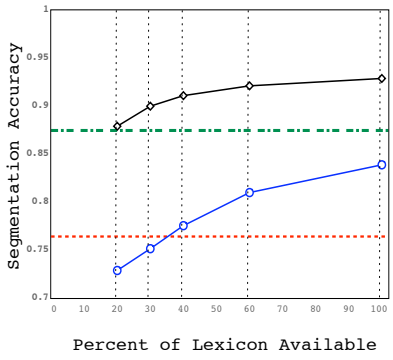
Figure 4(a) shows the learning curve of the lexicon-based approach in comparison to the trie and hybrid models. Because the size of the lexicon is so small, it is not until 40% lexicon usage that the lexicon-based approach significantly outperforms the prior. At 100% lexicon usage, the lexicon-based segmentation method achieves an accuracy of 83.9%, while the trie model alone achieves 87.5% accuracy. The hybrid trie backoff model is the clear winner, with an average accuracy of 92.9% at its maximum, and at all test points outperforms the standalone trie model.

5.2 Iraqi Arabic

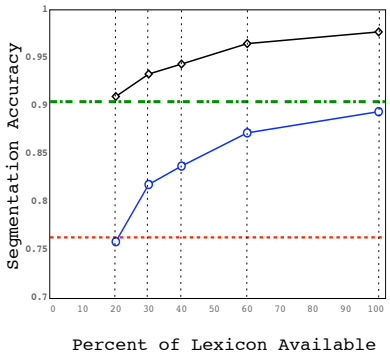
The results for Iraqi Arabic, shown in Figure 4(b), are quite similar to those from Levantine Arabic. The larger lexicon and corpus for this dialect lead to the higher accuracy rates. The trie model achieves an accuracy of 90.4%. Also, the hybrid trie model outperforms the standalone lexicon model by a range of 10% across the range of lexicon sizes – a roughly 50% error rate reduction.

Application to MT

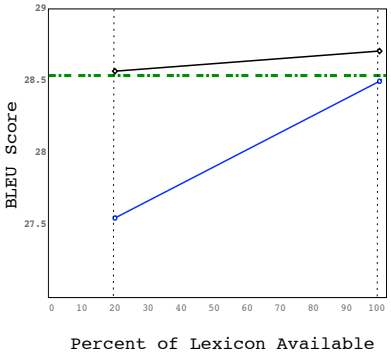
We apply morpheme segmentation using the lexicon, trie, and hybrid models to the Iraqi Arabic side of the English/Iraqi Arabic parallel corpus and record MT accuracy measured by BLEU score. In these experiments, we also compare against a baseline state-of-the-art phrase-based statistical MT system, as discussed in (Och and Ney, 2004) using GIZA++ word alignment training (Och and Ney,



(a) Learning curves for Levantine Arabic.



(b) Learning curves for Iraqi Arabic.



(c) BLEU Scores for English/Iraqi Arabic using the lexicon, trie, and hybrid models for segmentation. No morpheme segmentation yields a BLEU score of **26.07**.

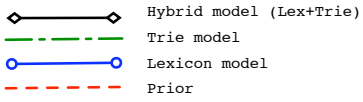


Figure 4. Learning curves for Iraqi and Levantine Arabic with trie-model and prior performance. Accuracies are averaged over all morphemes.

2000) with no morpheme segmentation.

Figure 4(c) shows graphically the results of performing morpheme segmentation before MT with the Lexicon, Trie, and Hybrid models. Accuracy is measured twice for the Lexicon and Hybrid models: once using the full lexicon and once at 20%. Table 4 gives numeric figures with confidence intervals.

The baseline model with no morpheme segmentation has a total training vocabulary size of 29,238 words, with 7.03% unknown words at translation time. The Trie model cuts the number of unknown words by more than half to 3.48% with a total vocabulary size of 16,878. The Hybrid model yields a further improvement to 3.01% unknown words and a total vocabulary size of 15,906. This is a 45.6% reduction from the original 29,238 unique word types handled by the baseline system.

6 Discussion

We have presented a minimally supervised model for morpheme segmentation with performance that exceeds a standard lexicon-match-based approach, even when trained on a small amount of data. Table 5 gives an illustrative set of examples. When the trie-based method is used as a backoff model for a small to moderate sized lexicon, the performance gains are better still. This hybrid approach consistently outperforms the standalone lexicon model by an average of 10% for both Iraqi and Levantine Arabic. This is a roughly 50% error rate reduction over the standard lexicon approach.

We have also shown that morpheme segmenta-

Model	BLEU	Confidence Interval
Baseline	26.07	24.79 - 27.42
20% Lexicon	27.55	26.36 - 28.91
Full Lexicon	28.50	27.22 - 29.82
Trie	28.54	27.18 - 29.87
Hybrid (20% Lexicon)	28.57	27.20 - 29.92
Hybrid (Full Lexicon)	**28.71**	27.43 - 30.06

Table 4. *BLEU scores with 95% confidence intervals corresponding to the sample points on the graph of Figure 4(c).*

Trie Model	Lexicon Model	Inflected Analysis	Uninflected Analysis
✓/0.6942	✓	**b+ syArp**	bsyArp
✓/0.6038	✘	b+ TAryAt	**bTAryAt**
✓/0.8110	✘	**Al+ mATwrAt**	AlmATwrAt
✓/0.7918	✘	Al+ ly	**Ally**
✓/0.7375	✓	**Ax +y**	Axy
✓/0.7568	✘	mksyk +y	**mksyky**

Table 5. Correct and incorrect model decisions for Iraqi Arabic morpheme segmentation. A ✓ denotes a correct segmentation decision; a ✘ denotes an incorrect decision. The decisions for the Trie model are accompanied by its associated probability score of making the correct analysis.

tion, when applied to a highly inflected language like Iraqi Arabic, yields a significant improvement in BLEU score and reduction in unknown words. Thus, morpheme segmentation for low-resource languages like dialectal Arabic helps to mitigate the data sparsity problem for statistical machine translation.

References

Altoma, S. J., "The Problem of Diglossia in Arabic: A Comparative Study of classical and Iraqi Arabic", *Harvard Middle Eastern Monograph Series*, Cambridge, MA., 1969.

Buckwalter, T., "Buckwalter Arabic Morphological Analyzer Version 2.0", *Linguistic Data Consortium*, catalog number LDC2004L02, 2004.

Cucerzan, S. and Yarowsky, D., "Language Independent Named Entity Recognition Combining Morphological and Contextual Evidence", In *Proceedings of the Joint SIGDAT Conference on EMNLP and VLC*, 1999.

Diab, M., Hacioglu, K., and Jurafsky, D., "Automatic tagging of Arabic text: From raw text to base phrase chunks", In *Proceedings of HLT/NAACL-2004*, 2004.

Habash, N., Rambow, O., and Kiraz, G. "Morphological Analysis and Generation for Arabic Dialects.", In *Proceedings of the Workshop on Computational Approaches to Semitic Languages* at the 43rd Annual Meeting of the ACL, 2005.

Habash, N. and Rambow, O., "Arabic Tokenization, Morphological Analysis, and Part-of-Speech Tagging in One Fell Swoop", In *Proceedings of the 43rd Annual Meeting of the ACL*, 2005.

Lee, Y.-S., Papineni, O., Roukos, S., Emam, O., Hassan, H. "Language Model Based Arabic Word Segmentation", In *Proceedings of the 41st Annual Meeting of the ACL*, 2003.

Lee, Y.-S. "Morphological Analysis for Statistical Machine Translation", In *Proceedings of HLT-NAACL 2004 Companion Volume*, pages 57-60, 2004.

Och, F. J. and Ney, H., "Improved statistical alignment models", In *Proceedings of the 38th Annual Meeting of the ACL*, pages 440-447, 2000.

Och, F. J. and Ney H. , "The alignment template approach to statistical machine translation", *Computational Linguistics*, 30:417-449, 2004.

Maamouri, M., Bies A., Buckwalter, T., "The Penn Arabic Treebank: Building a Large-Scale Annotated Arabic Corpus", In *NEMLAR Conference on Arabic Language Resources and Tools*, 2004.

Maamouri, M., Graff, D., Hubert, J., Cieri, C., Buckwalter, T., "Dialectal Arabic Orthography-based Transcription & CTS Levantine Arabic Collection", *Linguistic Data Consortium*, catalog number LDC2005E77, 2004.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J., "BLEU: a Method for Automatic Evaluation of Machine Translation", In *Proceedings of the 40th Annual Meeting of the ACL*, pages 311-318, 2002.

Riesa, J., Mohit, M., Knight, K., Marcu, D. "Building an English-Iraqi Arabic Machine Translation System for Spoken Utterances with Limited Resources", In *Proceedings of Interspeech 2006 - ICSLP*, 2006.

Wicentowski, R., "Modeling and Learning Multilingual Inflectional Morphology in a Minimally Supervised Framework", Ph.D. thesis, Johns Hopkins University, 2002.