Arabic Preprocessing Schemes for Statistical Machine Translation

Nizar Habash
Center for Computational Learning Systems
Columbia University
habash@cs.columbia.edu

Fatiha Sadat
Institute for Information Technology
National Research Council of Canada
fatiha.sadat@cnrc-nrc.gc.ca

Abstract

In this paper, we study the effect of different word-level preprocessing decisions for Arabic on SMT quality. Our results show that given large amounts of training data, splitting off only proclitics performs best. However, for small amounts of training data, it is best to apply English-like tokenization using part-of-speech tags, and sophisticated morphological analysis and disambiguation. Moreover, choosing the appropriate preprocessing produces a significant increase in BLEU score if there is a change in genre between training and test data.

1 Introduction

Approaches to statistical machine translation (SMT) are robust when it comes to the choice of their input representation: the only requirement is consistency between training and evaluation.[1] This leaves a wide range of possible preprocessing choices, even more so for morphologically rich languages such as Arabic. We use the term "preprocessing" to describe various input modifications that can be applied to raw training and evaluation texts for SMT to make them suitable for model training and decoding, including different kinds of tokenization, stemming, part-of-speech (POS) tagging and lemmatization. We refer to a specific kind of preprocessing as a "scheme" and differentiate it from the "technique" used to obtain it. Since we wish to study the effect of word-level preprocessing, we do not utilize any syntactic information. We define the word (and by extension its morphology) to be limited to written Modern Standard Arabic (MSA) strings separated by white space, punctuation and numbers. Thus, some prepositional particles and conjunctions are considered part of the word morphology.

In this paper, we report on an extensive study of the effect on SMT quality of six preprocessing schemes[2], applied to text disambiguated in three different techniques and across a learning curve. Our results are as follows: (a) for large amounts of training data, splitting off only proclitics performs best; (b) for small amount of training data, following an English-like tokenization and using part-of-speech tags performs best; (c) suitable choice of preprocessing yields a significant increase in BLEU score if there is little training data and/or there is a change in genre between training and test data; (d) sophisticated morphological analysis and disambiguation help significantly in the absence of large amounts of data.

Section 2 presents previous relevant research. Section 3 presents some relevant background on Arabic linguistics to motivate the schemes discussed in Section 4. Section 5 presents the tools and data sets used, along with the results of our experiments. Section 6 contains a discussion of the results.

2 Previous Work

The anecdotal intuition in the field is that reduction of word sparsity often improves translation quality. This reduction can be achieved by increasing training data or via morphologically driven preprocessing (Goldwater and McClosky, 2005). Recent publications on the effect of morphology on SMT quality focused on morphologically rich languages such as German (Nießen and Ney, 2004); Spanish, Catalan, and Serbian (Popović and Ney, 2004); and Czech (Goldwater and McClosky, 2005). They all studied

[1]This paper is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-06-C-0023. Any opinions, findings and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of DARPA. We thank Roland Kuhn, George Forster, Mona Diab, Owen Rambow, and Martin Jansche for helpful discussions.

[2]We conducted several additional experiments that we do not report on here for lack of space but we reserve for a separate technical report.

Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL, pages 49–52,
New York, June 2006.

the effects of various kinds of tokenization, lemmatization and POS tagging and show a positive effect on SMT quality. Specifically considering Arabic, Lee (2004) investigated the use of automatic alignment of POS tagged English and affix-stem segmented Arabic to determine appropriate tokenizations. Her results show that morphological preprocessing helps, but only for the smaller corpora. As size increases, the benefits diminish. Our results are comparable to hers in terms of BLEU score and consistent in terms of conclusions. We extend on previous work by experimenting with a wider range of preprocessing schemes for Arabic, by studying the effect of morphological disambiguation (beyond POS tagging) on preprocessing schemes over learning curves, and by investigating the effect on different genres.

3 Arabic Linguistic Issues

Arabic is a morphologically complex language with a large set of morphological features. These features are realized using both concatenative (affixes and stems) and templatic (root and patterns) morphology with a variety of morphological and phonological adjustments that appear in word orthography and interact with orthographic variations. Certain letters in Arabic script are often spelled inconsistently which leads to an increase in both sparsity (multiple forms of the same word) and ambiguity (same form corresponding to multiple words). For example, variants of Hamzated Alif, أ or إ are often written without their Hamza (ء): ا. Another example is the optionality of diacritics in Arabic script. We assume all of the text we are using is undiacritized.

Arabic has a set of attachable clitics to be distinguished from inflectional features such as gender, number, person and voice. These clitics are written attached to the word and thus increase its ambiguity. We can classify three degrees of cliticization that are applicable in a strict order to a word base:

`[CONJ+ [PART+ [Al+ BASE +PRON]]]`

At the deepest level, the BASE can have a definite article (Al+ *the*)[3] or a member of the class of pronominal enclitics, +PRON, (e.g. +hm *their/them*). Next comes the class of particle proclitics (PART+): l+ *to/for*, b+ *by/with*, k+ *as/such* and s+ *will/future*. Most shallow is the class of conjunction proclitics (CONJ+): w+ *and* and f+ *then*.

[3]Arabic transliterations are provided in the Buckwalter transliteration scheme (Buckwalter, 2002).

These phenomena highlight two issues related to preprocessing: First, ambiguity in Arabic words is an important issue to address. To determine whether a clitic or feature should be split off or abstracted off requires that we determine that said feature is indeed present in the word we are considering in context – not just that it is possible given an analyzer or, worse, because of regular expression matching. Secondly, once a specific analysis is determined, the process of splitting off or abstracting off a feature must be clear on what the form of the resulting word is to be. For example, the word كتبتهم *ktbthm* has two possible readings (among others) as *their writers* or *I wrote them*. Splitting off the pronominal clitic *+hm* without normalizing the *t* to *p* in the nominal reading leads to the coexistence of two forms of the noun: *ktbp* and *ktbt*. This increased sparsity is only worsened by the fact that the second form is also the verbal form (thus increased ambiguity).

4 Preprocessing: Schemes and Techniques

A scheme is a specification of the form of preprocessed output; whereas a technique is the method used to create such output. We examine six different schemes and three techniques.

4.1 Preprocessing Techniques

The different techniques chosen illustrate three degrees of linguistic knowledge dependence. The first is very light and cheap. The second is more expensive, requiring the use of a morphological analyzer. And the third is yet more expensive than the second; it is a disambiguation system that requires an analyzer and a disambiguated training corpus.

- REGEX is the baseline technique. It is simply greedy regular expression matching to modify strings and/or split off prefix/suffix substrings that look like clitics indicated by specific schemes. REGEX cannot be used with complex schemes such as EN and MR (see Section 4.2).
- BAMA, Buckwalter Arabic Morphological Analyzer (Buckwalter, 2002), is used to obtain possible word analyses. Using BAMA prevents incorrect greedy REGEX matches. Since BAMA produces multiple analyses, we always select one in a consistent arbitrary manner (first in a sorted list of analyses).
- MADA, The Morphological Analysis and Disambiguation for Arabic tool, is an off-the-shelf resource for Arabic disambiguation (Habash and

Table 1: The Different Preprocessing Schemes (with **MADA** Technique)

Input	wsynhY	Alr}ys	jwlth	bzyArp	AlY	trkyA.	
Gloss	and will fi nish	the president	tour his	with visit	to	Turkey	.
English	The president will fi nish his tour with a visit to Turkey.						
ST	wsynhY	Alr}ys	jwlth	bzyArp	AlY	trkyA	.
D1	w+ synhy	Alr}ys	jwlth	bzyArp	<lY	trkyA	.
D2	w+ s+ ynhy	Alr}ys	jwlth	b+ zyArp	<lY	trkyA	.
D3	w+ s+ ynhy	Al+ r}ys	jwlp +P_{3MS}	b+ zyArp	<lY	trkyA	.
MR	w+ s+ y+ nhy	Al+ r}ys	jwl +p +h	b+ zyAr +p	<lY	trkyA	.
EN	w+ s+ >nhY$_{VBP}$ +S_{3MS}	Al+ r}ys$_{NN}$	jwlp$_{NN}$ +P_{3MS}	b+ zyArp$_{NN}$	<lY$_{IN}$	trkyA$_{NNP}$	.

Rambow, 2005). MADA selects among BAMA analyses using a combination of classifiers for 10 orthogonal dimensions, including POS, number, gender, and pronominal clitics.

For BAMA and MADA, applying a preprocessing scheme involves moving features (as specified by the scheme) out of the chosen word analysis and regenerating the word without the split off features (Habash, 2004). The regeneration guarantees the normalization of the word form.

4.2 Preprocessing Schemes

Table 1 exemplifies the effect of the different schemes on the same sentence.

• **ST**: Simple Tokenization is the baseline preprocessing scheme. It is limited to splitting off punctuations and numbers from words and removing any diacritics that appear in the input. This scheme requires no disambiguation.

• **D1, D2, and D3**: Decliticizations. D1 splits off the class of conjunction clitics (*w+* and *f+*). D2 splits off the class of particles (*l+*, *k+*, *b+* and *s+*) beyond D1. Finally D3 splits off what D2 does in addition to the definite article (*Al+*) and all pronominal clitics.

• **MR**: Morphemes. This scheme breaks up words into stem and affixival morphemes.

• **EN**: English-like. This scheme is intended to minimize differences between Arabic and English. It decliticizes similarly to D3; however, it uses lexeme and English-like POS tags instead of the regenerated word and it indicates the pro-dropped verb subject explicitly as a separate token.

5 Experiments

We use the phrase-based SMT system, Portage (Sadat et al., 2005). For training, Portage uses IBM word alignment models (models 1 and 2) trained in both directions to extract phrase tables. Maximum phrase size used is 8. Trigram language models are implemented using the SRILM toolkit (Stolcke, 2002). Decoding weights are optimized using Och's algorithm (Och, 2003) to set weights for the four components of the log-linear model: language model, phrase translation model, distortion model, and word-length feature. The weights are optimized over the BLEU metric (Papineni et al., 2001). The Portage decoder, Canoe, is a dynamic-programming beam search algorithm, resembling the algorithm described in (Koehn, 2004a).

All of the training data we use is available from the Linguistic Data Consortium (LDC). We use an Arabic-English parallel corpus of about 5 million words for translation model training data.[4] We created the English language model from the English side of the parallel corpus together with 116 million words from the English Gigaword Corpus (LDC2005T12) and 128 million words from the English side of the UN Parallel corpus (LDC2004E13). English preprocessing comprised down-casing, separating punctuation from words and splitting off "'s". Arabic preprocessing was varied using the proposed schemes and techniques. Decoding weight optimization was done on 200 sentences from the 2003 NIST MT evaluation test set. We used two different test sets: (a) the 2004 NIST MT evaluation test set (MT04) and (b) the 2005 NIST MT evaluation test set (MT05). MT04 is a mix of news, editorials and speeches, whereas MT05, like the training data, is purely news. We use the evaluation metric BLEU-4 (Papineni et al., 2001).

We conducted all possible combinations of schemes and techniques discussed in Section 4 with different training corpus sizes: 1%, 10% and 100%. The results of the experiments are summarized in

[4]The parallel text includes Arabic News, eTIRR, English translation of Arabic Treebank, and Ummah.

Table 2: Results

	MT04									MT05								
	MADA			BAMA			REGEX			MADA			BAMA			REGEX		
	1	10	100	1	10	100	1	10	100	1	10	100	1	10	100	1	10	100
ST	9.4	22.9	34.6	9.4	22.9	34.6	9.4	22.9	34.6	11.2	27.7	37.8	11.2	27.7	**37.8**	11.2	27.7	37.8
D1	13.1	26.9	36.1	12.9	26.5	35.6	11.4	25.5	34.8	14.9	29.8	37.3	14.5	29.6	37.0	13.2	29.5	38.5
D2	14.2	27.7	**37.1**	13.7	27.9	**36.2**	12.0	25.5	**35.8**	16.3	30.2	**38.6**	15.5	31.0	**37.8**	13.4	**29.8**	**38.7**
D3	16.5	**28.7**	34.3	15.9	**28.3**	34.2	**13.6**	**26.1**	34.0	17.7	**31.0**	36.0	17.3	**31.1**	35.3	**14.7**	28.8	36.1
MR	11.6	27.5	34.4	14.2	27.5	33.4	n/a	n/a	n/a	12.7	29.6	35.9	15.7	29.5	34.3	n/a	n/a	n/a
EN	**17.5**	28.4	34.5	**16.3**	27.9	34.0	n/a	n/a	n/a	**18.3**	30.4	36.0	**17.6**	30.4	34.8	n/a	n/a	n/a

Table 2. All reported scores must have over 1.1% BLEU-4 difference to be significant at the 95% confidence level for 1% training. For all other training sizes, the difference must be over 1.7% BLEU-4. Error intervals were computed using bootstrap resampling (Koehn, 2004b).

6 Discussion

Across different schemes, **EN** performs the best under scarce-resource condition; and **D2** performs best under large-resource condition. Across techniques and under scarce-resource conditions, MADA is better than BAMA which is better than REGEX. Under large-resource conditions, this difference between techniques is statistically insignificant, though it's generally sustained across schemes.

The baseline for MT05, which is fully in news genre like training data, is considerably higher than MT04 (mix of genres). To investigate the effect of different schemes and techniques on different genres, we isolated in MT04 those sentences that come from the editorial and speech genres. We performed similar experiments as reported above on this subset of MT04. We found that the effect of the choice of the preprocessing technique+scheme was amplified. For example, MADA+**D2** (with 100% training) on non-news improved the system score 12% over the baseline **ST** (statistically significant) as compared to 2.4% for news only.

Further analysis shows that combination of output from all six schemes has a large *potential* improvement over all of the different systems, suggesting a high degree of complementarity. For example, a 19% improvement in BLEU score (for MT04 under MADA with 100% training) (from 37.1 in **D2** to 44.3) was found from an oracle combination created by selecting for each input sentence the output with the highest sentence-level BLEU score.

7 Future Work

We plan to study additional variants that these results suggest may be helpful. In particular, we plan to include more syntactic knowledge and investigate combination techniques at the sentence and subsentence levels.

References

T. Buckwalter. 2002. Buckwalter Arabic Morphological Analyzer. Linguistic Data Consortium. (LDC2002L49).

S. Goldwater and D. McClosky. 2005. Improving Statistical MT through Morphological Analysis. In *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

N. Habash. 2004. Large Scale Lexeme Based Arabic Morphological Generation. In *Proc. of Traitement Automatique du Langage Naturel*.

N. Habash and O. Rambow. 2005. Tokenization, Morphological Analysis, and Part-of-Speech Tagging for Arabic in One Fell Swoop. In *Proc. of the Association for Computational Linguistics (ACL)*.

P. Koehn. 2004a. Pharaoh: a Beam Search Decoder for Phrase-based Statistical Machine Translation Models. In *Proc. of the Association for Machine Translation in the Americas*.

P. Koehn. 2004b. Statistical Signifi cance Tests For Machine Translation Evaluation. In *Proc. of EMNLP*.

Y. Lee. 2004. Morphological Analysis for Statistical Machine Translation. In *Proc. of the North American Chapter of ACL*.

F. Och. 2003. Minimum Error Rate Training in Statistical Machine Translation. In *Proc. of ACL*.

K. Papineni, S. Roukos, T. Ward, and W. Zhu. 2001. Bleu: a Method for Automatic Evaluation of Machine Translation. Technical Report RC22176(W0109-022), IBM Research.

M. Popović and H. Ney. 2004. Towards the Use of Word Stems and Suffi xes for Statistical Machine Translation. In *Proc. of the Conference on Language Resources and Evaluation*.

S. Nießen and H. Ney. 2004. Statistical Machine Translation with Scarce Resources Using Morpho-syntactic Information. *Computational Linguistics*, 30(2).

F. Sadat, H. Johnson, A. Agbago, G. Foster, R. Kuhn, J. Martin, and A. Tikuisis. 2005. Portage: A Phrase-based Machine Translation System. In *Proc. of ACL Workshop on Building and Using Parallel Texts*.

Andreas Stolcke. 2002. Srilm - An Extensible Language Modeling Toolkit. In *Proc. of International Conference on Spoken Language Processing*.