

Dialectal to Standard Arabic Paraphrasing to Improve Arabic-English Statistical Machine Translation

Wael Salloum and Nizar Habash

Center for Computational Learning Systems

Columbia University

{wael, habash}@ccls.columbia.edu

Abstract

This paper is about improving the quality of Arabic-English statistical machine translation (SMT) on dialectal Arabic text using morphological knowledge. We present a light-weight rule-based approach to producing Modern Standard Arabic (MSA) paraphrases of dialectal Arabic out-of-vocabulary (OOV) words and low frequency words. Our approach extends an existing MSA analyzer with a small number of morphological clitics, and uses transfer rules to generate paraphrase lattices that are input to a state-of-the-art phrase-based SMT system. This approach improves BLEU scores on a blind test set by 0.56 absolute BLEU (or 1.5% relative). A manual error analysis of translated dialectal words shows that our system produces correct translations in 74% of the time for OOVs and 60% of the time for low frequency words.

1 Introduction

Much work has been done on Modern Standard Arabic (MSA) natural language processing (NLP) and machine translation (MT). In comparison, research on dialectal Arabic (DA), the unstandardized spoken varieties of Arabic, is still lacking in NLP in general and MT in particular. In this paper we address the issue of MT out-of-vocabulary (OOV) terms and low frequency terms in highly dialectal Arabic text.

We present a light-weight rule-based approach to producing MSA morphological paraphrases of DA OOV words and low frequency words. However, we don't do lexical translation. Our approach extends an existing MSA analyzer to two DA varieties (Levantine and Egyptian) with less than 40 morphologi-

cal clitics. We use 11 morphological transfer rules to generate paraphrase lattices that are input to a state-of-the-art phrase-based statistical MT (SMT) system. Our system improves BLEU scores on a blind test set by 0.56 absolute BLEU (or 1.5% relative). A manual error analysis of translated dialectal words shows that our system produces correct translations in 74% of the time for OOVs and 60% of the time for low frequency words.

The rest of this paper is structured as follows: Section 2 is related work, Section 3 presents linguistic challenges and motivation, Section 4 details our approach and Section 5 presents results evaluating our approach under a variety of conditions.

2 Related Work

Dialectal Arabic NLP Much work has been done in the context of MSA NLP (Habash, 2010). Specifically for Arabic-to-English SMT, the importance of tokenization using morphological analysis has been shown by many researchers (Lee, 2004; Zollmann et al., 2006; Habash and Sadat, 2006). In contrast, research on DA NLP is still in its early stages: (Kilany et al., 2002; Kirchhoff et al., 2003; Duh and Kirchhoff, 2005; Habash and Rambow, 2006; Chiang et al., 2006). Several researchers have explored the idea of exploiting existing MSA rich resources to build tools for DA NLP, e.g., Chiang et al. (2006) built syntactic parsers for DA trained on MSA treebanks. Such approaches typically expect the presence of tools/resources to relate DA words to their MSA variants or translations. Given that DA and MSA do not have much in terms of parallel corpora, rule-based methods to translate DA-to-MSA

Analyzer (BAMA), for instance, produces an average of 12 analyses per word. Moreover, some letters in Arabic 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), e.g., variants of Hamzated Alif, أ \hat{A} or إ \check{A} , are often written without their Hamza (ء): أ A ; and the Alif-Maqsurā (or dotless Ya) ي y and the regular dotted Ya ي y are often used interchangeably in word final position (Kholy and Habash, 2010). Arabic complex morphology and ambiguity are handled using tools for disambiguation and tokenization (Habash and Rambow, 2005; Diab et al., 2007). For our SMT system, we preprocess the Arabic text so that it is tokenized in the Penn Arabic Treebank tokenization (Maamouri et al., 2004), Alif/Ya normalized and undiacritized. These measures have an important effect on reducing overall OOV rate (Habash, 2008).

3.2 Dialectal Arabic Challenges

Contemporary Arabic is in fact a collection of varieties: MSA, which has a standard orthography and is used in formal settings, and DAs, which are commonly used informally and with increasing presence on the web, but which do not have standard orthographies. There are several varieties of DA which primarily vary geographically, e.g., Levantine Arabic, Egyptian Arabic, etc. DAs differ from MSA phonologically, morphologically and to some lesser degree syntactically. The differences between MSA and DAs have often been compared to Latin and the Romance languages (Habash, 2006). The morphological differences are most noticeably expressed in the use of clitics and affixes that do not exist in MSA. For instance, the Levantine Arabic equivalent of the MSA example above is $w+H+y-ktb-w+hA$ وحيككتبوها ‘and they will write it’. The optionality of vocalic diacritics helps hide some of the differences resulting from vowel changes; compare the diacritized forms: Levantine $wHayuktubuwhA$ and MSA $wasayaktubuwnahA$.

All of the NLP challenges of MSA described above are shared by DA. However, the lack of standard orthographies for the dialects and their numerous varieties pose new challenges. Additionally, DAs are rather impoverished in terms of available

tools and resources compared to MSA; e.g., there is very little parallel DA-English corpora and almost no MSA-DA parallel corpora. The number and sophistication of morphological analysis and disambiguation tools in DA is very limited in comparison to MSA (Duh and Kirchhoff, 2005; Habash and Rambow, 2006; Abo Bakr et al., 2008). MSA tools cannot be effectively used to handle DA: Habash and Rambow (2006) report that less than two-thirds of Levantine verbs can be analyzed using an MSA morphological analyzer.

3.3 Dialectal Arabic OOVs

We analyzed the types of OOVs in our dev set against our large system (see Section 5) with an eye for dialectal morphology. The token OOV rate is 1.51% and the type OOV rate is 7.45%; although the token OOV rate may seem small, it corresponds to almost one third of all sentences having one OOV at least (31.48%). In comparison with MSA test sets, such as NIST MTEval 2006’s token OOV rate of 0.8% (and 3.42% type OOV rate), these numbers are very high specially given the size of training data. Out of these OOVs, 25.9% have MSA readings or are proper nouns. The rest, 74.1%, are dialectal words. We classified the dialectal words into two types: words that have MSA-like stems and dialectal affixational morphology (affixes/clitics) and those that have dialectal stem and possibly dialectal morphology. The former set accounts for almost half of all OOVs (49.7%) or almost two thirds of all dialectal OOVs. In this paper we only target dialectal affixational morphology cases as they are the largest class involving dialectal phenomena that do not require extension to our stem lexica. The morphological coverage of the analyzer we use, ALMOR, which itself uses the BAMA databases is only 21% of all the OOV words. Our analyzer, ADAM, presented in Section 4.2, improves coverage substantially.

It is important to note that a word can be invocabulary (INV) but not have a correct possible translation in the phrase table. Some of these words may be of such low frequency that their various possible translations simply do not appear in the training data. Others may have a frequent MSA reading and an infrequent/unseen DA reading (or vice versa).

4 Approach

Our basic approach to address the issue of translational OOVs is to provide rule-based paraphrases of the source language words into words and phrases that are INV. The paraphrases are provided as alternatives in an input lattice to the SMT system. This particular implementation allows this approach to be easily integrated with a variety of SMT systems. The alternatives include different analyses of the same original word and/or translations into MSA. We focus on the question of Arabic dialects, although the approach can be extended to handle low frequency MSA words also that may have been mis-tokenized by the MSA preprocessing tools. As mentioned above, we only report in this work on dialect morphology translation to MSA and we leave lemma/word translation to future work. We identify four distinct operations necessary for this approach and evaluate different subsets of them in Section 5.

1. **Selection.** Identify the words to handle, e.g., OOVs or low frequency words.
2. **Analysis.** Produce a set of alternative analyses for each word.
3. **Transfer.** Map each analysis into one or more target analyses.
4. **Generation.** Generate properly tokenized forms of the target analyses.

The core steps of analysis-transfer-generation are similar to generic transfer-based MT (Dorr et al., 1999). In essence our approach can be thought of as a mini-rule-based system that is used to hybridize an SMT system (Simard et al., 2007; Sawaf, 2010).

4.1 Selection

The most obvious set of words to select for paraphrasing is the phrase-table OOV words. We identify them by comparing each word in the source text against all phrase-table singletons. Another set of words to consider includes low frequency words (DA or MSA), which are less likely to be associated with good phrase-table translations. We compute the frequency of such words against the original training data. We further extend the idea of frequency-based selection to typed-frequency selection in which we consider different frequency cut-offs for different

types of words (MSA or DA). Evaluation and more details are presented in Section 5.3.

4.2 Analysis

Whereas much work has been done on MSA morphological analysis (Al-Sughaiyer and Al-Kharashi, 2004), a small handful of efforts have targeted the creation of dialectal morphology systems (Kilany et al., 2002; Habash and Rambow, 2006; Abo Bakr et al., 2008). In this section, we present a new dialectal morphological analyzer, ADAM, built as an extension to an already existing MSA analyzer. We only focus on extensions that address dialectal affixes and clitics, as opposed to stems, which we plan to address in future work. This approach to extending an MSA analyzer is similar to work done by Abo Bakr et al. (2008) and it contrasts as rather a shallow/quick-and-dirty solution compared to other more demanding efforts on building dialectal analyzers from scratch, such as the MAGEAD system (Habash and Rambow, 2006; Altantawy et al., 2011).

4.2.1 ADAM: Analyzer for Dialectal Arabic Morphology

ADAM is built on the top of BAMA database (Buckwalter, 2004) as used in the ALMOR morphological analyzer/generator (Habash, 2007), which is the rule-based component of the MADA system for morphological analysis and disambiguation of Arabic (Habash and Rambow, 2005; Roth et al., 2008). The ALMOR system presents analyses as lemma and feature-value pairs including clitics.

The BAMA databases contain three tables of Arabic stems, complex prefixes and complex suffixes² and three additional tables with constraints on matching them. MSA, according to the BAMA databases, has 1,208 complex prefixes and 940 complex suffixes, which correspond to 49 simple prefixes/proclitics and 177 simple suffixes/enclitics, respectively. The number of combinations in prefixes is a lot bigger than in suffixes, which explains the different proportions of complex affixes to simple affixes.

We extended the BAMA database through a

²We define a *complex prefix* as the full sequence of prefixes/proclitics that may appear at the beginning of a word. *Complex suffixes* are defined similarly.

Dialect Word	وماحيكتبلو <i>wmAHyktblw</i> ‘And he will not write for him’					
Analysis	Proclitics			[Lemma & Features]	Enclitics	
	w+ conj+ and+	mA+ neg+ not+	H+ fut+ will+	yktb [katab IV subj:3MS voice:act] he writes	+l +prep +for	+w +pron _{3MS} +him
Transfer	Word 1		Word 2	Word 3		
	Proclitics	[Lemma & Features]	[Lemma & Features]	[Lemma & Features]	Enclitic	
	conj+ and+	[lan] will not	[katab IV subj:3MS voice:act] he writes	[li] for	+pron _{3MS} +him	
Generation	w+	ln	yktb	l	+h	
MSA Phrase	ولن يكتب له <i>wln yktb lh</i> ‘And he will not write for him’					

Figure 1: An example illustrating the analysis-transfer-generation steps to translate a word with dialectal morphology into its MSA equivalent phrase.

set of rules that add new Levantine/Egyptian dialectal affixes and clitics by copying and extending existing MSA affixes/clitics. For instance, the dialectal future proclitic +ح *H+* ‘will’ has a similar behavior to the standard Arabic future particle +س *s+*. As such, an extension rule would create a copy of each occurrence of the MSA prefix and replace it with the dialectal prefix. The algorithm that uses this rule to extend the BAMA database adds the prefix *Ha/FUT_PART* and many other combinations involving it, e.g., *wa/PART+Ha/FUT_PART+ya/IV3MS*, and *fa/CONJ+Ha/FUT_PART+na/IV1P*. We reserve discussion of other more complex mappings with no exact MSA equivalence to a future publication on ADAM.

The rules (89 in total) introduce 11 new dialectal proclitics (plus spelling variants and combinations) and 27 dialectal enclitics (again, plus spelling variants and combinations). ADAM’s total of simple prefixes and suffixes increases to 60 (22% increase) and 204 (15% increase) over BAMA, respectively. The numbers for complex prefixes and suffixes increase at a faster rate to 3,234 (168% increase) and (142% increase), respectively.

As an example of ADAM output, consider the second set of rows in Figure 1, where a single analysis is shown.

4.2.2 ADAM performance

We conducted an analysis of ADAM’s behavior over the OOV set analyzed in Section 3.3. Whereas ALMOR (before ADAM) only produces analyzes for 21% of all the OOV words, ADAM covers almost

63%. Among words with dialectal morphology, ADAM’s coverage is 84.4%. The vast majority of the unhandled dialectal morphology cases involve a particular Levantine/Egyptian suffix +ش *+š* ‘not’. We plan to address these cases in the future. In about 10% of all the analyzed words, ADAM generates alternative dialectal readings to supplement existing ALMOR MSA analyses, e.g., *بكتب* *bktb* has an MSA (and coincidentally dialectal) analysis of ‘with books’ and ADAM also generates the dialectal only analysis ‘I write’.

4.3 Transfer

In the transfer step, we map ADAM’s dialectal analyses to MSA analyses. This step is implemented using a set of transfer rules (TR) that operate on the lemma and feature representation produced by ADAM. The TRs can change clitics, features or lemma, and even split up the dialectal word into multiple MSA word analyses. Crucially the input and output of this step are both in the lemma and feature representation (Habash, 2007). A particular analysis may trigger more than one rule resulting in multiple paraphrases. This only adds to the fan-out which started with the original dialectal word having multiple analyses.

Our current system uses 11 rules only, which were determined to handle all the dialectal clitics added in ADAM. As more clitics are added in ADAM, more TRs will be needed. As examples, two TRs which lead to the transfer output shown in the third set of rows in Figure 1 can be described as follows:³

³All of our rules are written in a declarative form, which

- if the dialectal analysis shows future and negation proclitics, remove them from the word and create a new word, the MSA negative-future particle لن *ln*, to precede the current word and which inherits all proclitics preceding the future and negation proclitics.
- if the dialectal analysis shows the dialectal indirect object enclitic, remove it from the word and create a new word to follow the current word; the new word is the preposition +ل *l+* with an enclitic pronoun that matches the features of the indirect object.

In the current version evaluated in this paper, we always provide a lower-scored back-off analysis that removes all dialectal clitics as an option.

4.4 Generation

In this step, we generate Arabic words from all analyses produced by the previous steps. The generation is done using the general tokenizer TOKAN (Habash, 2007) to produce Arabic Treebank (ATB) scheme tokenizations. TOKAN is used in the baseline system to generate tokenizations for MSA from morphologically disambiguated input in the same ATB scheme (see Section 5.1). The various generated forms are added in the lattices, which are then input to the SMT system.

5 Evaluation on Machine Translation

5.1 Experimental Setup

We use the open-source Moses toolkit (Koehn et al., 2007) to build two phrase-based SMT systems trained on two different data conditions: a medium-scale MSA-only system trained using a newswire (MSA-English) parallel text with 12M words on the Arabic side (LDC2007E103) and a large-scale MSA/DA-mixed system (64M words on the Arabic side) trained using several LDC corpora including some limited DA data. Both systems use a standard phrase-based architecture. The parallel corpus is word-aligned using GIZA++ (Och and Ney, 2003). Phrase translations of up to 10 words are extracted in the Moses phrase table. The language model for both systems is trained on the English

may be complicated to explain given the allotted space, as such we present only the functional description of the TRs.

side of the large bitext augmented with English Gigaword data. We use a 5-gram language model with modified Kneser-Ney smoothing. Feature weights are tuned to maximize BLEU on the NIST MTEval 2006 test set using Minimum Error Rate Training (Och, 2003). This is only done on the baseline systems.

For all systems, the English data is tokenized using simple punctuation-based rules. The Arabic side is segmented according to the Arabic Treebank tokenization scheme (Maamouri et al., 2004) using the MADA+TOKAN morphological analyzer and tokenizer (Habash and Rambow, 2005) – v3.1 (Roth et al., 2008). The Arabic text is also Alif/Ya normalized (Habash, 2010). MADA-produced Arabic lemmas are used for word alignment.

Results are presented in terms of BLEU (Papineni et al., 2002), NIST (Doddington, 2002) and METEOR (Banerjee and Lavie, 2005) metrics.⁴ However, all optimizations were done against the BLEU metric. All evaluation results are case insensitive.

All of the systems we present use the lattice input format to Moses (Dyer et al., 2008), including the baselines which do not need them. We do not report on the non-lattice baselines, but in initial experiments we conducted, they did not perform as well as the degenerate lattice version.

The Devtest Set Our devtest set consists of sentences containing at least one non-MSA segment (as annotated by LDC)⁵ in the Dev10 audio development data under the DARPA GALE program. The data contains broadcast conversational (BC) segments (with three reference translations), and broadcast news (BN) segments (with only one reference, replicated three times). The data set contained a mix of Arabic dialects, with Levantine Arabic being the most common variety. The particular nature of the devtest being transcripts of audio data adds some challenges to MT systems trained on primarily written data in news genre. For instance, each of the source and references in the devtest set contained over 2,600 *uh*-like speech effect words (*uh/ah/oh/eh*), while the baseline translation system we used only generated 395. This led to severe

⁴We use METEOR version 1.2 with four match modules: exact, stem, wordnet, and paraphrases.

⁵<http://www.ldc.upenn.edu/>

brevity penalty by the BLEU metric. As such, we removed all of these speech effect words in the source, references and our MT system output. Another similar issue was the overwhelming presence of commas in the English reference compared to the Arabic source: each reference had about 14,200 commas, while the source had only 64 commas. Our MT system baseline predicted commas in less than half of the reference cases. Similarly we remove commas from the source, references, and MT output. We do this to all the systems we compare in this paper. However, even with all of this preprocessing, the length penalty was around 0.95 on average in the large system and around 0.85 on average in the medium system. As such, we report additional BLEU sub-scores, namely the unigram and bigram precisions (Prec-1 and Prec-2, respectively), to provide additional understanding of the nature of our improvements.

We split this devtest set into two sets: a development set (dev) and a blind test set (test). We report all our analyses and experiments on the dev set and reserve the test set for best parameter runs at the end of this section. The splitting is done randomly at the document level. The dev set has 1,496 sentences with 32,047 untokenized Arabic words. The test set has 1,568 sentences with 32,492 untokenized Arabic words.

5.2 Handling Out-of-Vocabulary Words

In this section, we present our results on handling OOVs in our baseline MT system following the approach we described in Section 4. The results are summarized in Table 1. The table is broken into two parts corresponding to the large and medium systems. Each part contains results in BLEU, Prec-1 (unigram precision), Prec-2 (bigram precision), NIST and METEOR metrics. The performance of the large system is a lot better than the medium system in all experiments. Some of the difference is simply due to training size; however, another factor is that the medium system is trained on MSA only data while the large system has DA in its training data.

We compare the baseline system (first row) to two methods of OOV handling through dialectal paraphrase into MSA. The first method uses the ADAM morphological analyzer and generates directly skip-

ping the transfer step to MSA. Although this may create implausible output for many cases, it is sufficient for some, especially through the system’s natural addressing of orthographic variations. This method appears in Table 1 as ADAM Only. The second method includes the full approach as discussed in Section 4, i.e., including the transfer step.

The use of the morphological analyzer only method (ADAM Only) yields positive improvements across all metrics and training data size conditions. In the medium system, the improvement is around 0.42% absolute BLEU (or 2.1% relative). The large system improves by about 0.34% absolute BLEU (or almost 1% relative). Although these improvements are small, they are only accomplished by targeting a part of the OOV words (about 0.6% of all words).

The addition of transfer rules leads to further modest improvements in both large and medium systems according to BLEU; however, the NIST and METEOR metrics yield negative results in the medium system. A possible explanation for the difference in behavior is that paraphrase-based approaches to MT often suffer in smaller data conditions since the paraphrases they map into may themselves be OOVs against a limited system. Our transfer approach also has a tendency to generate longer paraphrases as options, which may have lead to more fragmentation in the METEOR score algorithm. In terms of BLEU scores, the full system (analysis and transfer) improves over the baseline on the order of 0.5% BLEU absolute. The relative BLEU score in the large and medium systems are 1.24% and 2.54% respectively.

All the systems in Table 1 do not drop unhandled OOVs, thus differing from the most common method of “handling” OOV, which is known to game popular MT evaluation metrics such as BLEU (Habash, 2008). In fact, if we drop OOVs in our baseline system, we get a higher BLEU score of 36.36 in the large system whose reported baseline gets 36.16 BLEU. That said, our best result with OOV handling produces a higher BLEU score (36.61) which is a nice result for doing the right thing and not just deleting problem words. All differences in BLEU scores in the large system are statistically significant above the 95% level. Statistical significance is computed using paired bootstrap resampling (Koehn, 2004).

System	Large (64M words)					Medium (12M words)				
	BLEU	Prec-1	Prec-2	NIST	METEOR	BLEU	Prec-1	Prec-2	NIST	METEOR
Baseline	36.16	74.56	45.04	8.9958	52.59	20.09	63.69	30.89	6.0039	40.85
ADAM Only	36.50	74.79	45.22	9.0655	52.95	20.51	64.37	31.22	6.1994	41.80
ADAM+ Transfer	36.61	74.85	45.37	9.0825	53.02	20.60	64.70	31.48	6.1740	41.77

Table 1: Results for the dev set under large and medium training conditions. The baseline is compared to using dialectal morphological analysis only and analysis plus transfer to MSA. BLEU and METEOR scores are presented as percentages.

System	Large (64M words)				
	BLEU	Prec-1	Prec-2	NIST	METEOR
Baseline	36.16	74.56	45.04	8.9958	52.59
ADAM+ Transfer	36.61	74.85	45.37	9.0825	53.02
+ Freq $x \leq 10$	36.71	74.89	45.50	9.0821	52.97
+ Freq $x_{MSA} \leq 10$	36.62	74.86	45.38	9.0816	52.96
+ Freq $x_{DIAMSA} \leq 13$	36.66	74.86	45.43	9.0836	53.01
+ Freq $x_{DIA} \leq 45$	36.73	75.00	45.57	9.0961	53.03
+ Freq $x_{MSA} \leq 10 + x_{DIAMSA} \leq 13 + x_{DIA} \leq 45$	36.78	74.96	45.61	9.0926	52.96

Table 2: Results for the dev set under large training condition, varying the set of words selected for MSA paraphrasing.

5.3 Extending Word Selection

Following the observation that some dialectal words may not pose a challenge to SMT since they appear frequently in training data, while some MSA words may be challenging since they are infrequent, we conduct a few experiments that widen the set of words selected for DA-MSA paraphrasing. We report our results on the large data condition only. Results are shown in Table 2. The baseline and best system from Table 1 are repeated for convenience.

We consider two types of word-selection extensions beyond OOVs. First, we consider frequency-based selection, where all words with less than or equal to a frequency of x are considered for paraphrasing in addition to being handled in the system’s phrase table. Many low frequency words actually end up being OOVs as far as the phrase table is concerned since they are not aligned properly or at all by GIZA++. Secondly we consider a typed-frequency approach, where different frequency values are considered depending on whether a word is MSA only, dialect only or has both dialect and MSA readings. We determine MSA words to be those that have ALMOR analyses but no new ADAM analyses. Dialect-only words are those that have ADAM analyses but no ALMOR analyses. Finally, dialect/MSA words are those that have ALMOR analyses and get more

dialect analyses through ADAM. The intuition behind the distinction is that problematic MSA only words may be much less frequent than problematic dialectal words.

We conducted a large number of experiments to empirically determine the best value for x in the frequency-based approach and x_{MSA} , x_{DIA} , and x_{DIAMSA} for the typed frequency approach. For the typed frequency approach, we took a greedy path to determine optimal values for each case and then used the best results collectively. Our best values are presented in Table 2. Both frequency-based approaches improve over the best results of only targeting OOVs. Further more, the fine-tuned typed frequency approach even yields further improvements leading to 0.62% absolute BLEU improvement over the baseline (or 1.71% relative). This score is statistically significant against the baseline and the ADAM+Transfer system as measured using paired bootstrap resampling (Koehn, 2004).

5.4 Blind Test Results

We apply our two basic system variants and best result with typed frequency selection to the blind test set. The results are shown in Table 3. The test set overall has slightly higher scores than the dev set, suggesting it may be easier to translate relatively.

System	Large (64M words)				
	BLEU	Prec-1	Prec-2	NIST	METEOR
Baseline	37.24	75.12	46.40	9.1599	52.93
ADAM Only	37.63	75.40	46.59	9.2414	53.39
ADAM+ Transfer	37.71	75.46	46.70	9.2472	53.41
+ Freq $x_{MSA} \leq 10 + x_{DIAMSA} \leq 13 + x_{DIA} \leq 45$	37.80	75.47	46.82	9.2578	53.44

Table 3: Results for the blind test set under large training condition, comparing our best performing settings.

All of our system variants improve over the baseline and show the same rank in performance as on the dev set. Our best performer improves over the baseline by 0.56 absolute BLEU (or 1.5% relative). The relative increase in Prec-2 is higher than in Prec-1 suggesting perhaps that some improvements are coming from better word order.

5.5 Manual Error Analysis

We conduct two manual error analyses comparing the baseline to our best system. First we compare the baseline system to our best system applied only to OOVs. Among all 656 OOV tokens (1.51%) in our dev set we attempt to handle 417 tokens (0.96%) (i.e., 63.57% of possible OOVs) which could possibly affect 320 sentences (21.39%); however, we only see a change in 247 sentences (16.51%). We took a 50-sentence sample from these 247 sentences (our sample is 20%). We classified every occurrence of an OOV into not handled (the output has the OOV word), mistranslated (including deleted), or corrected (the output contains the correct translation); we focused on adequacy rather than fluency in this analysis. Table 4 presents some examples from the analysis set illustrating different behaviors. Among the OOVs in the sample (total 68 instances), 22% are not handled. Among the handled cases, we successfully translate 74% of the cases. Translation errors are mostly due to spelling errors, lexical ambiguity or proper names. There are no OOV deletions. This analysis suggests that our results reflect the correctness of the approach as opposed to random BLEU bias due to sentence length, etc.

In the second manual error analysis, we compare two systems to help us understand the effect of handling low frequency (LF) words: (a) our best system applied only to OOVs [OOV], and (b) our best system applied to OOVs and LF words [OOV+LF]. For LF words only (as compared to OOVs), we attempt

to handle 669 tokens (1.54%) which could possibly affect 489 sentence (32.69%); however, we see a change in only 268 sentences (17.91%) (as compared to the OOV handling system). We took a 50-sentence sample from these sentences in the dev set where the output of the two systems is different (total 268 sentences; our sample is 19%). We classified each LF word into mistranslated or correct, and we annotated each case as dialectal, MSA, or tokenization error. Among the LF words in the sample (total 64 instances), the [OOV+LF] system successfully translated 55% of the cases while the [OOV] system successfully translated 50% of the cases. Overall, 11% of all LF words in our sample are due to a tokenization error, 34% are MSA, and 55% are dialectal. Among dialectal cases, the [OOV+LF] system successfully translated 60% of the cases while the [OOV] system successfully translated 42% of the cases. Among MSA cases, the [OOV+LF] system successfully translates 55% of the cases while the [OOV] system successfully translate 64% of the cases. The conclusion here is that (a) the majority of LF cases handled are dialectal and (b) the approach to handle them is helpful; however (c) the LF handling approach may hurt MSA words overall. Table 5 presents some examples from the analysis set illustrating different behaviors.

6 Conclusion and Future Work

We presented a light-weight rule-based approach to producing MSA paraphrases of dialectal Arabic OOV words and low frequency words. The generated paraphrase lattices result in improved BLEU scores on a blind test set by 0.56 absolute BLEU (or 1.5% relative). In the future, we plan to extend our system’s coverage of phenomena in the handled dialects and on new dialects. We are interested in using ADAM to extend the usability of existing morphological disambiguation systems for MSA to the

Arabic	yṣny ṣn AlAzdHAmAt btstxdmwn ¹ AlbnšklAt ² ?
Reference	You mean for traffic jams you use ¹ the bicycles ² ?
Baseline	I mean, about the traffic btstxdmwn ¹ AlbnšklAt ² ?
OOV-Handle	I mean, about the traffic use ¹ AlbnšklAt ² ?
Arabic	nHnA bntAm ³ Anh fy hḏA Almwqf tbdA msyrḥ jdydḥ slmyḥ mTlwbḥ lAlmnTqḥ .
Reference	We hope ³ in this situation to start a new peace process that the region needs.
Baseline	We bntAm ³ that in this situation start a new march peaceful needed for the region.
OOV-Handle	We hope ³ that this situation will start a new march peaceful needed for the region.
Arabic	dktwr Anwr mAjd ṣṣqy ⁴ rḡys mrkz Alšrq AlAwsT lldrAsAt AlAstrAtyjyḥ mn AlryAD ...
Reference	Dr. Anwar Majid ' Ishqi ⁴ President of the Middle East Center for Strategic Studies from Riyadh ...
Baseline	Dr. anwar majed ṣṣqy ⁴ head of middle east center for strategic studies from riyadh ...
OOV-Handle	Dr. anwar majed love ⁴ , president of the middle east center for strategic studies from riyadh ...

Table 4: Examples of different results of handling OOV words. Words of interest are bolded. Superscript indexes are used to link the related words within each example. Words with index 1 and 3 are correctly translated; the word with index 2 is not handled; and the word with index 4 is an incorrectly translated proper name.

Arabic	... wḏlk HtṣAml mṣ Aljmyṣ ṣly hAlAsAs .
Reference	... and I shall therefore deal with everyone on this basis .
OOV	... and therefore dealt with everyone to think .
OOV+LF	... and therefore dealt with everyone on this basis .
Arabic	... tṣydown nfs Alkrḥ An lm ykn AswA ...
Reference	... repeat the same thing if not worse ...
OOV	... to re - the same if not worse ...
OOV+LF	... bring back the same if not worse ...

Table 5: Examples of different results of handling LF words. Words of interest are bolded. Both examples show a LF word mistranslated in the first system and successfully translated in the second system. The first examples shows a dialectal word while the second example shows an MSA word.

dialects, e.g., MADA. Furthermore, we want to automatically learn additional morphological system rules and transfer rules from limited available data (DA-MSA or DA-English) or at least use these resources to learn weights for the manually created rules.

Acknowledgments

This research was supported by the DARPA GALE program, contract HR0011-06-C-0022. Any opinions, findings, conclusions or recommendations expressed in this work are those of the authors and do not necessarily reflect the view of DARPA. We would like to thank Amit Abbi for help with the MT baseline. We also would like to thank John Makhoul, Richard Schwartz, Spyros Matsoukas, Rabih Zbib and Mike Kayser for helpful discussions and feedback and for providing us with the devtest data.

References

- Hitham Abo Bakr, Khaled Shaalan, and Ibrahim Ziedan. 2008. A Hybrid Approach for Converting Written Egyptian Colloquial Dialect into Diacritized Arabic. In *The 6th International Conference on Informatics and Systems, INFOS2008*. Cairo University.
- Rania Al-Sabbagh and Roxana Girju. 2010. Mining the Web for the Induction of a Dialectical Arabic Lexicon. In Nicoletta Calzolari, Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *LREC*. European Language Resources Association.
- Imad A. Al-Sughaiyer and Ibrahim A. Al-Kharashi. 2004. Arabic morphological analysis techniques: A comprehensive survey. *Journal of the American Society for Information Science and Technology*, 55(3):189–213.
- Mohamed Altantawy, Nizar Habash, and Owen Rambow. 2011. Fast Yet Rich Morphological Analysis. In *proceedings of the 9th International Workshop on Finite-State Methods and Natural Language Processing (FSMNL 2011)*, Blois, France.

- Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan.
- Tim Buckwalter. 2004. Buckwalter Arabic Morphological Analyzer Version 2.0. Linguistic Data Consortium, University of Pennsylvania. LDC Catalog No.: LDC2004L02, ISBN 1-58563-324-0.
- Chris Callison-Burch, Philipp Koehn, and Miles Osborne. 2006. Improved statistical machine translation using paraphrases. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 17–24.
- David Chiang, Mona Diab, Nizar Habash, Owen Rambow, and Safiullah Shareef. 2006. Parsing Arabic Dialects. In *Proceedings of the European Chapter of ACL (EACL)*.
- Mona Diab, Kadri Hacioglu, and Daniel Jurafsky. 2007. *Arabic Computational Morphology: Knowledge-based and Empirical Methods*, chapter Automated Methods for Processing Arabic Text: From Tokenization to Base Phrase Chunking. Springer.
- George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Human Language Technology*, pages 128–132, San Diego.
- Bonnie J. Dorr, Pamela W. Jordan, and John W. Benoit. 1999. A Survey of Current Research in Machine Translation. In M. Zekowitz, editor, *Advances in Computers, Vol. 49*, pages 1–68. Academic Press, London.
- Jinhua Du, Jie Jiang, and Andy Way. 2010. Facilitating translation using source language paraphrase lattices. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, EMNLP’10*, pages 420–429, Cambridge, Massachusetts.
- Kevin Duh and Katrin Kirchhoff. 2005. POS tagging of dialectal Arabic: a minimally supervised approach. In *Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages, Semitic ’05*, pages 55–62, Ann Arbor, Michigan.
- Christopher Dyer, Smaranda Muresan, and Philip Resnik. 2008. Generalizing word lattice translation. In *Proceedings of ACL-08: HLT*, Columbus, Ohio.
- Nizar Habash and Owen Rambow. 2005. Arabic Tokenization, Part-of-Speech Tagging and Morphological Disambiguation in One Fell Swoop. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL’05)*, pages 573–580, Ann Arbor, Michigan.
- Nizar Habash and Owen Rambow. 2006. MAGEAD: A Morphological Analyzer and Generator for the Arabic Dialects. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 681–688, Sydney, Australia.
- Nizar Habash and Fatiha Sadat. 2006. Arabic Preprocessing Schemes for Statistical Machine Translation. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 49–52, New York City, USA.
- Nizar Habash, Abdelhadi Soudi, and Tim Buckwalter. 2007. On Arabic Transliteration. In A. van den Bosch and A. Soudi, editors, *Arabic Computational Morphology: Knowledge-based and Empirical Methods*. Springer.
- Nizar Habash. 2006. On Arabic and its Dialects. *Multilingual Magazine*, 17(81).
- Nizar Habash. 2007. Arabic Morphological Representations for Machine Translation. In A. van den Bosch and A. Soudi, editors, *Arabic Computational Morphology: Knowledge-based and Empirical Methods*. Springer.
- Nizar Habash. 2008. Four Techniques for Online Handling of Out-of-Vocabulary Words in Arabic-English Statistical Machine Translation. In *Proceedings of ACL-08: HLT, Short Papers*, pages 57–60, Columbus, Ohio.
- Nizar Habash. 2010. *Introduction to Arabic Natural Language Processing*. Morgan & Claypool Publishers.
- Jan Hajič, Jan Hric, and Vladislav Kubon. 2000. Machine Translation of Very Close Languages. In *Proceedings of the 6th Applied Natural Language Processing Conference (ANLP’2000)*, pages 7–12, Seattle.
- Ahmed El Kholy and Nizar Habash. 2010. Techniques for Arabic Morphological Detokenization and Orthographic Denormalization. In *Workshop on Language Resources and Human Language Technology for Semitic Languages in the Language Resources and Evaluation Conference (LREC)*, Valletta, Malta.
- H. Kilany, H. Gadalla, H. Arram, A. Yacoub, A. El-Habashi, and C. McLemore. 2002. Egyptian Colloquial Arabic Lexicon. LDC catalog number LDC99L22.
- Katrin Kirchhoff, Jeff Bilmes, Sourin Das, Nicolae Duta, Melissa Egan, Gang Ji, Feng He, John Henderson, Daben Liu, Mohamed Noamany, Pat Schone, Richard Schwartz, and Dimitra Vergyri. 2003. Novel Approaches to Arabic Speech Recognition: Report from the 2002 Johns Hopkins Summer Workshop. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Hong Kong, China.

- P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic.
- Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the Empirical Methods in Natural Language Processing Conference (EMNLP'04)*, Barcelona, Spain.
- Shankar Kumar, Franz J. Och, and Wolfgang Macherey. 2007. Improving word alignment with bridge languages. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 42–50, Prague, Czech Republic.
- Young-Suk Lee. 2004. Morphological Analysis for Statistical Machine Translation. In *Proceedings of the 5th Meeting of the North American Chapter of the Association for Computational Linguistics/Human Language Technologies Conference (HLT-NAACL04)*, pages 57–60, Boston, MA.
- Mohamed Maamouri, Ann Bies, Tim Buckwalter, and Wigdan Mekki. 2004. The Penn Arabic Treebank: Building a Large-Scale Annotated Arabic Corpus. In *NEMLAR Conference on Arabic Language Resources and Tools*, pages 102–109, Cairo, Egypt.
- Preslav Nakov and Hwee Tou Ng. 2011. Translating from Morphologically Complex Languages: A Paraphrase-Based Approach. In *Proceedings of the Meeting of the Association for Computational Linguistics (ACL'2011)*, Portland, Oregon, USA.
- F. J. Och and H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.
- Franz Josef Och. 2003. Minimum Error Rate Training for Statistical Machine Translation. In *Proceedings of the 41st Annual Conference of the Association for Computational Linguistics*, pages 160–167, Sapporo, Japan.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, PA.
- Jason Riesa and David Yarowsky. 2006. Minimally Supervised Morphological Segmentation with Applications to Machine Translation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas (AMTA06)*, pages 185–192, Cambridge, MA.
- Ryan Roth, Owen Rambow, Nizar Habash, Mona Diab, and Cynthia Rudin. 2008. Arabic Morphological Tagging, Diacritization, and Lemmatization Using Lexeme Models and Feature Ranking. In *Proceedings of ACL-08: HLT, Short Papers*, pages 117–120, Columbus, Ohio.
- Hassan Sawaf. 2010. Arabic dialect handling in hybrid machine translation. In *Proceedings of the Conference of the Association for Machine Translation in the Americas (AMTA)*, Denver, Colorado.
- Michel Simard, Nicola Ueffing, Pierre Isabelle, and Roland Kuhn. 2007. Rule-based translation with statistical phrase-based post-editing. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 203–206, Prague, Czech Republic.
- Masao Utiyama and Hitoshi Isahara. 2007. A comparison of pivot methods for phrase-based statistical machine translation. In *HLT-NAACL*, pages 484–491.
- Xiaoheng Zhang. 1998. Dialect MT: a case study between Cantonese and Mandarin. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, ACL '98, pages 1460–1464, Montreal, Canada.
- Andreas Zollmann, Ashish Venugopal, and Stephan Vogel. 2006. Bridging the Inflection Morphology Gap for Arabic Statistical Machine Translation. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 201–204, New York City, USA.