REMOOV: A Tool for Online Handling of Out-of-Vocabulary Words in Machine Translation

Nizar Habash

Center for Computational Learning Systems
Columbia University
habash@ccls.columbia.edu

Abstract

REMOOV is a tool for online handling of out-of-vocabulary (OOV) words in statistical machine translation. REMOOV employs four techniques. Spelling expansion and morphological expansion are used to produce alternative in-vocabulary (INV) forms of OOV words. Dictionary term expansion and proper name transliteration produce target translations directly. These techniques can be used to expand the phrase table utilized in decoding or as part of an input/output lattice expansion. Results of using REMOOV show a consistent improvement over a state-of-the-art baseline. This paper describes the different components and parameters of the REMOOV tool.

1. Introduction

REMOOV is a tool for online handling of Out-of-Vocabulary (OOV) words in phrase-based statistical machine translation. REMOOV employs four techniques to reuse or extend phrase tables online: morphological expansion (MORPHEX), spelling expansion (SPELLEX), dictionary word expansion (DICTEX) and proper name transliteration (TRANSEX) (Habash, 2008). In this paper, we describe the REMOOV tool and its techniques in some detail to complement a previous publication (Habash, 2008). We also discuss different ways of using REMOOV beyond the specific experiments previously published.

The paper is structured as follows. Section 2. presents previous and related research. Section 3. presents some relevant background on Arabic linguistic issues and profiles OOVs in Arabic-English machine translation. Section 4. provides a high-level description of the REMOOV tool. Sections 5. through 8. describe the four different techniques used in REMOOV.

2. Previous and Related Work

Much work in machine translation (MT) has shown that orthographic and morpho-syntactic preprocessing of the training and test data reduces data sparsity and OOV rates. This is especially true for languages with rich morphology such as Spanish, Catalan, and Serbian (Popović and Ney, 2004) and Arabic (Sadat and Habash, 2006). But even in improved models that reduce sparsity, OOVs due to unseen proper names, spelling errors and less common morphological forms are still a problem. The most common "solution" for OOVs is deleting them from the output – thus gaming precision-based evaluation metrics such as BLEU (Papineni et al., 2002). Some previous approaches anticipate OOV words that are potentially morphologically related to in-vocabulary (INV) words (Yang and Kirchhoff, 2006). Habash and Metsky (2008) describe a morphological expansion technique for handling OOVs that does not require a morphological analyzer. Vilar et al. (2007) address spelling-variant OOVs in MT through online re-tokenization into letters and combination with a word-based system. There is much work on name transliteration and its integration in larger MT systems (Hassan and Sorensen, 2005; Hermjakob et al., 2008). Okuma et al. (2007) describe a novel dictionary-based technique for translating OOV words in statistical MT.

REMOOV builds on previously reported results in Habash (2008), where several evaluations were conducted to test the value of four techniques for handling OOVs in a phrase-based statistical MT system. The results in Habash (2008) showed that the techniques used in REMOOV improve over a state-of-the-art baseline by over 2.7% (relative BLEU score). This is significant given that the increase was obtained only by addressing OOVs (2.9% of all tokens). Error analysis showed that, in 60% of the time, OOV handling successfully produces acceptable output.

Although the experiments in (Habash, 2008) were conducted using a specific preprocessing scheme, the Arabic Treebank scheme (Sadat and Habash, 2006) and a specific phrase-based MT system (Koehn, 2004); REMOOV can be configured for other preprocessing schemes and can be used with other MT systems. The specific implementation we discuss here was done in the context of Arabic-English MT; however, the techniques can be used with other language pairs provided that the necessary technique-specific resources are available.

3. Out-of-Vocabulary Words in Arabic-English MT

Arabic Linguistic Issues Arabic orthography and morphology present many challenges that motivate our work. Orthographically, we distinguish four challenges for Arabic processing. First, Arabic script uses *optional* diacritics (less than 1.5% of tokens typically bear at least one diacritic). Although the presence of diacritics is helpful to human readers as a disambiguation aid; their inconsistent use makes them unreliable for automatic processing. Second, certain letters in Arabic script are often spelled inconsistently, e.g., variants of Hamzated Alef, أ Â[1] or إ Ă, are often

[1]Our system internally uses the Buckwalter Arabic transliteration scheme (Buckwalter, 2004); however, examples of Arabic text in this document are presented in the Habash-Soudi-Buckwalter (HSB) transliteration scheme (Habash et al., 2007). This scheme extends Buckwalter's scheme to increase its readability while maintaining a 1-to-1 correspondence with Arabic orthography as represented in standard encodings of Arabic, such as Unicode. The following are the only differences from Buckwal-

written without Hamza: ا *A*. Third, Arabic's alphabet uses *obligatory* dots to distinguish different letters (e.g., ب *b*, ت *t* and ث *θ*). Each letter base is ambiguous two ways on average. Added or missing dots are often seen in spelling errors. Finally, although the Arabic script is a mostly connective cursive script, there are a few letters that do not connect to the letters that follow them, e.g., ا *A*, ر *r*, د *d* and و *w*. This leads to the presence of a tiny word-internal space that sometimes is confused for a word separator. As a result, some words may be broken into two parts or more; and incorrect words are made up of two or more words.

Morphologically, Arabic is a rich language with a large set of morphological features such as gender, number, person and voice. Additionally, Arabic has a set of very common clitics that are written attached to the word, e.g., the conjunction و+ *w+* 'and', the preposition ل+ *l+* 'for/to' and the pronominal clitic هم+ *+hm* 'them/their'. For example, the word, ولمكتباتهم *wlmktbAthm* 'and for their libraries' can be analyzed to و+ل+مكتبة+ات+هم *w+l+mktbħ+At+hm* 'and+for+library+plural+their'.

Some of these challenges can be addressed by removing all diacritics, normalizing Alef and Ya forms,[2] and tokenizing Arabic text (Sadat and Habash, 2006). For example, tokenization in the Arabic Treebank tokenization scheme reduces OOV rates by 59% relative to raw text producing a token OOV rate of 2.89%. The rest of the challenges such as spelling errors and morphological variations can be addressed using REMOOV. REMOOV has no preference to a specific tokenization scheme. Five of the different schemes described in (Sadat and Habash, 2006) are currently supported (D1, D2, D3, TB and raw text).

Profile of OOV words in Arabic-English MT A study described in (Habash, 2008) showed the following part-of-speech distribution of OOV cases: proper nouns (40%), nouns (26.4%), verbs (19.3%) and adjectives (14.3%). The proper noun OOVs come from different origins including Arabic, Hebrew, English, French, and Chinese. For non-proper nouns, the OOV words were often the less common morphological variants of INV words, such as the nominal dual form. Spelling errors are also responsible for some of the OOV cases.

The different techniques in REMOOV address these different cases in different ways. Proper name transliteration is primarily handled by TRANSEX. However, an OOV with a different spelling of an INV name can be handled by SPELLEX. Morphological variants are handled primarily by MORPHEX and DICTEX, but since some morphological variations involve small changes in lettering, SPELLEX may contribute too, e.g., كتبتم *ktbt**m*** 'you [masc.pl.] wrote' and كتبتن *ktbt**n*** 'you [fem.pl.] wrote'.

ter's scheme (which is indicated in parentheses): Ā آ (|), Â أ (>), ŵ ؤ (&), Ǎ إ (<), ŷ ئ (}), ħ ة (p), θ ث (v), ð ذ (*), š ش ($), Ď ظ (Z), ς ع (E), γ غ (g), ý ى (Y), ã ً (F), ũ ٌ (N), ĩ ٍ (K), á ـَ (').

[2]For Alef normalization, آأإ *Ā/Â/Ǎ* are mapped to ا *A*. For Ya normalization, ى *ý* is mapped to ي *y*.

4. REMOOV

REMOOV's basic approach to handling OOVs is to extend the phrase table with possible translations of these OOVs. In the MORPHEX and SPELLEX techniques, REMOOV matches the OOV word with an INV word that is a possible variant of the OOV word. Phrases associated with the INV token in the phrase table are "recycled" to create new phrases in which the INV word is replaced with the OOV word. The translation weights of the INV phrase are used "as is" in the new phrase. The default setting in REMOOV is to limit the added phrases to source-language unigrams and bigrams (determined empirically), but this is a configurable parameter. In the DICTEX and TRANSEX techniques, REMOOV maps OOV words to new target translations and adds them to the phrase table. Although phrase-table expansion is the default method in REMOOV, the generated OOV-to-INV and OOV-to-target mappings can be used for lattice expansion in MT systems that use lattice inputs (Dyer et al., 2008).

The REMOOV tool uses a configuration file that allows users to specify which techniques to use and in which order. For instance, all techniques can be applied to all OOV words, or some techniques can be applied as back-off to other techniques. In particular, morphologically constrained techniques may not be able to produce an answer all the time; whereas spelling/surfacy techniques can almost always relate an OOV to some INV.

REMOOV requires an offline step, in which various data models are built from the phrase table. This is done to speed up the online process. The created models include: a list of all source-language words in the phrase table (the INV words), all morphological analyses associated with the INV words, and the morphological expansion rules used by the MORPHEX technique.

The different REMOOV techniques are described in the next four sections.

5. Morphology Expansion

In this technique, we match the OOV token with an INV token that is a possible morphological variant of the OOV token. For this to work, we need to be able to morphologically analyze the OOV word. OOV words that fail morphological analysis cannot be helped by this technique. The morphological matching process requires the words to be matched to agree in their lexeme (lemma) but have different inflectional features. We collect information on possible inflectional variations from the original phrase table itself. In an offline process, we cluster all the analyses of single word Arabic entries in our phrase table that (1) translate into the same English word and (2) have the same lexeme analysis. From these clusters we learn which morphological inflectional features in Arabic are irrelevant to English. We create a rule set of morphological inflection maps that we then use to relate analyses of OOV words to analyses of INV words. The following three automatically learned mapping rules exemplify what is captured well:

`[POS:N +GEN +INDEF]` ⇔ `[POS:N Al+ +ACC +DEF]`

`[POS:AJ +FEM +DU +ACC]` ⇔ `[POS:AJ +MASC +PL +NOM]`

`[POS:V +PV +S:3MS]` ⇔ `[POS:V +PV +S:3FS]`

The first rule states that genitive indefinite nouns in Arabic can be mapped to their accusative definite form with the

definite article clitic. The second rule states that feminine dual accusative nouns can map to the masculine plural nominative form. And the last rule states that perfective verbs conjugated for 3rd person masculine singular can be replaced with the 3rd person feminine singular form. The distinctions that Arabic makes in case, gender and dual/plural number are not always relevant to English. Since each mapping rule includes the full morphological inflectional vector of an Arabic word (minus clitics), the number of mapping rules can be quite large. Note that the learned rules will be different for different tokenization schemes. The examples presented here are for the Arabic Treebank tokenization scheme.

For nouns, the most common inflectional variation is the addition or deletion of the Arabic definite article *Al+*. This mapping allows the OOV words زبيدي *zbydy* 'Zubaydi' and الزلازل *AlzlAzl* 'the-earthquakes' to be related to الزبيدي *Alzbydy* 'Al-Zubaydi' and زلازل *zlAzl* 'earthquakes', respectively. Verbal inflectional variation include altering many of the values of the verbal word base such as number, gender, aspect, mood, etc. For instance, the OOV verb زرنا *zrnA* '(we) visited' can be related to زاروا *zArwA* '(they) visited', زار *zAr* '(he) visited' and زارت *zArt* '(she) visited'. This large set of mappings happens because English verbal morphology is quite impoverished compared to Arabic. We expect that using this technique on a morphologically richer language (such as French or Czech) would produce more restrictions.

Phrases associated with the INV token in the phrase table are used to create new phrases in which the INV token is replaced with the OOV token. The translation weights of the INV phrase are used "as is" in the new phrase. In the future we plan to investigate how to modify the weights using the probabilities of the learned rules. The three examples above produce the following phrases among others:

```
zbydy   ||| al-zubaydi   ||| (weights)
AlzlAzl ||| earthquakes  ||| (weights)
AlzlAzl ||| seismology   ||| (weights)
zrnA    ||| are visiting ||| (weights)
zrnA    ||| had visited  ||| (weights)
zrnA    ||| have visited ||| (weights)
```

6. Spelling Expansion

In SPELLEX, REMOOV matches the OOV token with an INV token that may be a possible correct spelling of the OOV token. In our current implementation, we consider five types of spelling correction involving one letter position only. We list them and exemplify them against the correctly spelled word فلسطيني *flsTyny* 'Palestinian':

- Letter deletion: فلسطني *flsTny*. A letter is deleted.
- Letter Insertion: فلسطيننني *flsTynny*. A letter is inserted.
- Letter inversion: فلطسيني *flTsyny*. Two letters are reversed in order.
- Letter substitution: قلسطيني *qlsTyny*. A letter is substituted by another letter. We allow a limited set of letter substitutions. The cases we consider include common letter shape alternations (e.g., ر/ز *r/z*, ب/ت/ث *b/t/θ* and ف/ق *f/q*), phonological alternations (e.g., س/ص *s/S*, د/ض *d/D* and ت/ط *t/T*) and dialectal variations (e.g., ق *q* alternating with ء/ك/ج *'/k/j*, and ك *k* alternating with تش *tš*). The list of substitutions can be easily modified in REMOOV.
- Word split: جارفلسطيني *jArflsTyny* → جار_فلسطيني *jAr_flsTyny* 'Palestinian neighbor [lit. neighbor_Palestinian]'. We allow adding a space to split the OOV word into two INV words.

We do not currently handle multiple types of spelling errors in the same word. Phrases associated with the INV tokens in the phrase table are used to create new phrases in which the INV tokens are replaced with their OOV variants. The translation weights of the INV phrases are used "as is" in the new phrase. For instance, in the letter inversion example above, فلطسيني *flTsyny* matching with the word فلسطيني *flsTyny* allows us to recycle its phrases and add them to the existing phrase table during translation. Here are some of the new phrases:

```
flTsyny ||| a palestinian ||| (weights)
flTsyny ||| of palestinian ||| (weights)
flTsyny ||| of palestinians ||| (weights)
flTsyny ||| of the palestinian ||| (weights)
flTsyny ||| palestinian , ||| (weights)
```

7. Dictionary Expansion

The DICTEX technique extends the phrase table with entries from a manually created dictionary, namely the glosses associated with the Buckwalter Arabic Morphological Analyzer (BAMA) output (Buckwalter, 2004). Common wisdom in statistical MT suggests that adding a dictionary to the training data rarely helps. What we do here is not simple addition of the uninflected lexemic entries. Instead we expand the English glosses of all the analyses that matched the OOV word to all their (English) possible surface forms. Given the large number of word forms associated with an Arabic lexeme and the often multiple English glosses, we run this process online as needed for OOV words. The newly generated pairs are assigned very low translation probabilities that do not interfere with the rest of the phrase table. All entries receive the same weights. During translation, the decoder will rank these entries only using the language model.

In the following example, the noun الزور *Alzwr* has the BAMA lexeme entry *zuwr* and is mapped to the English lexemes *falsehood/lie* among others. The verb علمنا ʕ*lmnA* is mapped to the lexeme entry ʕ*alim* which is glossed as *know/find_out* among others.

```
Alzwr ||| falsehood ||| (weights)
Alzwr ||| falsehoods ||| (weights)
Alzwr ||| lie ||| (weights)
Alzwr ||| lies ||| (weights)
ʕlmnA ||| know ||| (weights)
ʕlmnA ||| knew ||| (weights)
ʕlmnA ||| knows ||| (weights)
ʕlmnA ||| finds out ||| (weights)
```

ςlmnA ||| find out ||| (weights)
ςlmnA ||| found out ||| (weights)

8. Name Transliteration

The TRANSEX technique produces English transliteration hypotheses that assume the OOV is a proper name. The transliteration approach is rather simple. It uses the transliteration similarity measure described by Freeman et al. (2006) to select a best match from a large list of possible names in English. The list was collected from a large set of English corpora primarily using capitalization statistics. Since efficiency is a major concern for online processing, we constrain the search in our database of 280K entries using a *sounds-like* phonetic indexing algorithm called Double Metaphone[3] (DM) (Philips, 1990; Philips, 2000). DM is related to the well-known Soundex algorithm,[4] which maps similarly sounding proper names to a common fixed length code. DM keys collapse the phonetic consonants of a name into one of 14 *metaphones*. For example, Mark, Marco, Marick and Margo all map to MRK. The name Schwarzenegger is ambiguously mapped to XRSNKR and XFRTSNKR. Each entry in our name database has one or two associated metaphone keys. We created a loose mapping from Arabic to metaphones that outputs several possibilities for each word. The metaphones are only used to restrict the search. The final ranking of possible transliterations is solely based on the transliteration similarity measure. For example, the name باستور *bAstwr* is mapped to eight metaphone keys: PSTR, PSTAR, PASTR, PASTAR, FSTR, FSTAR, FASTR, FASTAR. The following are the top 20 possible transliterations produced by the system, listed alphabetically and by their similarity scores: (score of 1.00): Bastar, Bastyr, Bestar, Bester, Bestor, Pasteaur, Paster, Pasteur, Pastewr, Pastuer, Peaster, Postaer, Vestar, Vestaur, Vester, and (score of 0.86): Pastora, Pastore, Pastory, Pesatori, Pistore.

The newly generated pairs are assigned very low translation probabilities that do not interfere with the rest of the phrase table. Weights of entries are modulated using the similarity measure.

9. Conclusion and Future Plans

REMOOV is a tool for handling Out-of-Vocabulary words in MT through spelling expansion, morphological expansion, dictionary term expansion and proper name transliteration. This tool is publicly available for research purposes. Please contact the author for more information.

Although REMOOV was designed with MT in mind, we plan to explore other NLP tasks where OOV reduction is needed such as speech recognition. In the future, we plan to investigate how to improve each of the different techniques in REMOOV and explore better ways of automatically weighing the different generated hypotheses. Finally, we plan to extend REMOOV to other languages.

10. Acknowledgements

We would like to thank Owen Rambow and Ryan Roth for helpful discussions and support. This work has been funded by Defense Advanced Research Projects Agency Contract No. HR0011-06-C-0023 and Contract No. HR0011-08-C-0110. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of DARPA.

[3] http://search.cpan.org/dist/Text-DoubleMetaphone/

[4] http://www.archives.gov/genealogy/census/soundex.html

11. References

Tim Buckwalter. 2004. Buckwalter Arabic Morphological Analyzer Version 2.0. Linguistic Data Consortium, University of Pennsylvania, 2002. LDC Catalog No.: LDC2004L02.

Christopher Dyer, Smaranda Muresan, and Philip Resnik. 2008. Generalizing word lattice translation. In *Proceedings of the Association for Computational Linguistics (ACL-08)*, Columbus, OH.

A. Freeman, S. Condon, and C. Ackerman. 2006. Cross linguistic name matching in English and Arabic. In *Proceedings of the North American ACL (NAACL-06)*, New York City, NY.

Nizar Habash and Hayden Metsky. 2008. Automatic learning of morphological variations for handling out-of-vocabulary terms in Urdu-English machine translation. In *Proceedings of the Association for Machine Translation in the Americas (AMTA-08)*, Waikiki, HI.

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. 2008. Four Techniques for Online Handling of Out-of-Vocabulary Words in Arabic-English Statistical Machine Translation. In *Proceedings of ACL-08*.

Hany Hassan and Jeffrey Sorensen. 2005. An integrated approach for Arabic-English named entity translation. In *Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages*, Ann Arbor, MI.

Ulf Hermjakob, Kevin Knight, and Hal Daumé III. 2008. Name Translation in Statistical Machine Translation - Learning when to Transliterate. In *Proceedings of ACL-08*.

Philipp Koehn. 2004. Pharaoh: a Beam Search Decoder for Phrase-based Statistical Machine Translation Models. In *Proceedings of AMTA-04*.

H. Okuma, H. Yamamoto, and E. Sumita. 2007. Introducing translation dictionary into phrase-based SMT. In *Proceedings of the Machine Translation Summit (MT SUMMIT XI)*, Copenhagen, Denmark.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of ACL-02*, Philadelphia, PA.

Lawrence Philips. 1990. Hanging on the metaphone. *Computer Language Magazine*, 7(12).

Lawrence Philips. 2000. The double metaphone search algorithm. *C/C++ Users Journal*, June issue.

Maja Popović and Hermann Ney. 2004. Towards the Use of Word Stems and Suffixes for Statistical Machine Translation. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, Lisbon, Portugal.

Fatiha Sadat and Nizar Habash. 2006. Combination of Arabic Preprocessing Schemes for Statistical Machine Translation. In *Proceedings of ACL-06*, Sydney, Australia.

David Vilar, Jan-Thorsten Peter, and Hermann Ney. 2007. Can we translate letters? In *Proceedings of the ACL Workshop on Statistical Machine Translation*, Prague, Czech Republic.

Mei Yang and Katrin Kirchhoff. 2006. Phrase-based Backoff Models for Machine Translation of Highly Inflected Languages. In *Proceedings of Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, Trento, Italy.