Fuzzy Syntactic Reordering for Phrase-based Statistical Machine Translation

Jacob Andreas and **Nizar Habash** and **Owen Rambow**
Center for Computational Learning Systems
Columbia University
jda2129@columbia.edu
{habash,rambow}@ccls.columbia.edu

Abstract

The quality of Arabic-English statistical machine translation often suffers as a result of standard phrase-based SMT systems' inability to perform long-range re-orderings, specifically those needed to translate VSO-ordered Arabic sentences. This problem is further exacerbated by the low performance of Arabic parsers on subject and subject span detection. In this paper, we present two parse "fuzzification" techniques which allow the translation system to select among a range of possible S–V re-orderings. With this approach, we demonstrate a 0.3-point improvement in BLEU score (69% of the maximum possible using gold parses), and a corresponding improvement in the percentage of syntactically well-formed subjects under a manual evaluation.

1 Introduction

The question of how to effectively use phrase-based statistical machine translation (PSMT) to translate between language pairs which require long-range reordering has attracted a great deal of interest in recent years. The inability to capture long-range reordering behaviors is a weakness inherent in PSMT systems, which typically have only two mechanisms to control the reordering between source and target language: (1) distortion penalties, which penalize or forbid long-distance re-orderings in order to reduce the search space explored by the decoder, and (2) lexicalized reordering models, which capture the preferences of individual phrases to orient themselves monotonically, reversed with their preceding phrases or discontinuously. Because both of these mechanisms work at the phrase level, they have proven very effective at capturing short-range reordering behaviors, but unable to describe long range movements; in fact, the distortion penalty effectively causes the translation system to not prefer long-range re-orderings, even when they are assigned significantly higher probability by the language model.

The problem is particularly acute in translating from Arabic to English: Arabic sentences frequently exhibit a VSO ordering (both VSO and SVO are permitted in Arabic), while English permits only an SVO order. Past research has shown that verb anticipation and subject-span detection is a major source of error when translating from Arabic to English (Green et al., 2009; Bisazza and Federico, 2010). Unable to perform long-range reordering, PSMT frequently produces English sentences in which verbs precede their subjects (sometimes with "hallucinated" pronouns in front of them) or do not appear at all. Intuitively, better handling of these reorderings has the potential to improve both accuracy and fluency of translation.

In this paper, we present two parse fuzzification techniques which allow the translation system to select among a range of possible S–V re-orderings. With this approach, we demonstrate a 0.3-point improvement in BLEU score (69% of the maximum possible using gold parses), and a corresponding improvement in the percentage of syntactically well-formed subjects under a manual evaluation.

The rest of the paper is structured as follows. Section 2 gives a review of research on this topic. Section 3 motivates the approach discussed in Section 4.

Proceedings of the 6th Workshop on Statistical Machine Translation, pages 227–236,
Edinburgh, Scotland, UK, July 30–31, 2011.

Section 5 presents the results of a set of machine translation experiments using the automatic metrics BLEU (Papineni et al., 2002) and METEOR (Banerjee and Lavie, 2005), and a manual-evaluation of subject integrity. Section 6 discusses our conclusions and future plans.

2 Related Work

The general approach pursued in this paper—that of using pre-ordering to improve translation output– has been explored by many researchers. Most work has focused on automatically learning reordering rules (Xia and McCord, 2004; Habash, 2007b; Elming, 2008; Elming and Habash, 2009; Dyer and Resnik, 2010). Xia and McCord (2004) describe an approach for translation from French to English, where context-free constituency reordering rules are acquired automatically using source and target parses and word alignment. Elming (2008) and Elming and Habash (2009) use a large set of linguistic features to automatically learn reordering rules for English-Danish and English-Arabic; the rules are used to pre-order the input into a lattice of variant orders. Habash (2007b) learns syntactic reordering rules targeting Arabic-English word order differences and integrated them as deterministic preprocessing. He reports improvements in BLEU compared to phrase-based SMT limited to monotonic decoding, but these improvements do not hold with distortion. He hypothesizes that parse errors are responsible for lack of improvement. Dyer and Resnik (2010) use an input forest structure to represent word-order alternatives and learn models for long-range source reordering that maximize translation quality. Their results for Arabic-English are negative.

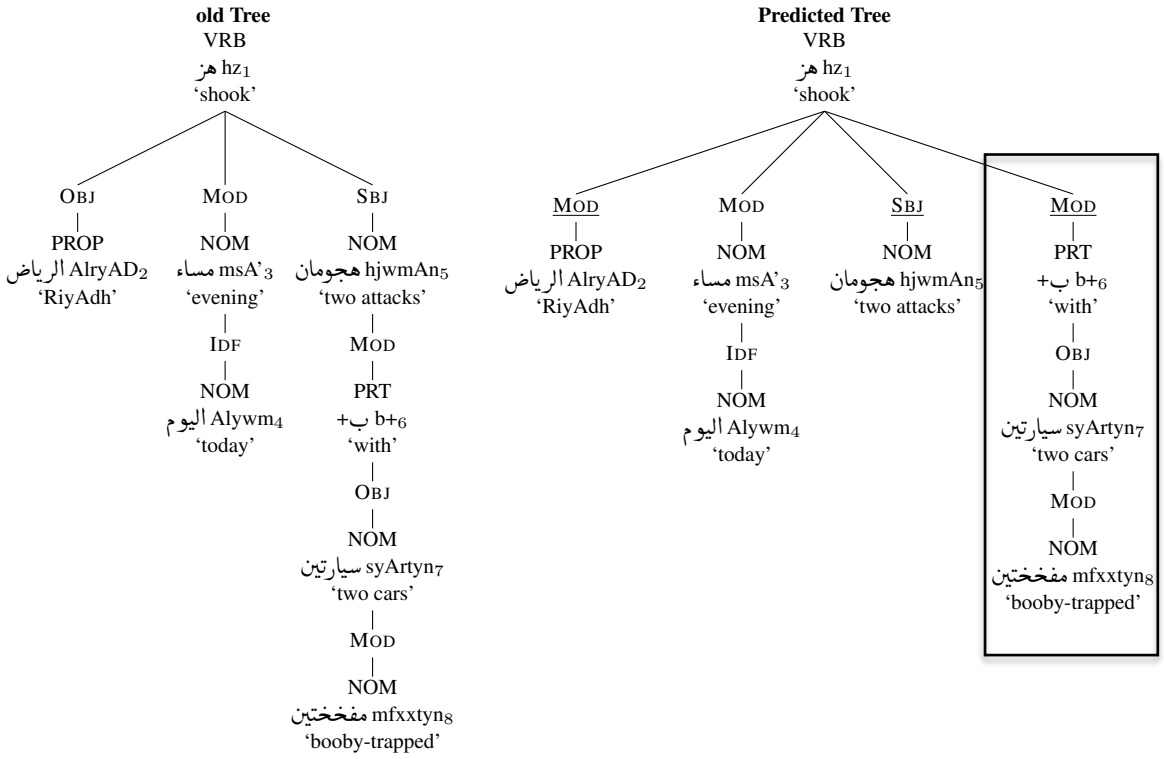
In contrast to these approaches, Collins et al. (2005) apply six *manually* defined transformations to German parse trees which yield an improvement on a German-English translation task. In this paper, we follow Collins et al. (2005) and restrict ourselves to handcrafted rules (in our case, actually a single over-generating rule) motivated by linguistic understanding.

One major concern not addressed in any of the aforementioned research on syntax-based reordering is the fact that the quality of parsers for many languages is still quite poor. Collins et al. (2005), for example, assume that the parse trees they use are correct. While the state-of-the-art in English parsing is fairly good (though far from perfect), this is not the case in other languages, where parsing shows substantial error rates. Moreover, when attempting to reorder so as to bring the source text more grammatically in line with the target language, a bad parse can be disastrous: moving parts of the sentence that shouldn't be moved, and introducing more distortion error than it is able to correct. To address the problem of noisy parse data, Bisazza and Federico (2010) identify the subject using a chunker, then *fuzzify* it, creating a lattice in which the translation system has a choice of several different paths, corresponding to re-orderings of different subject spans.

In investigating syntax-based reordering for Arabic specifically, Carpuat et al. (2010) show that a syntax-driven reordering of the training data only for the purpose of alignment improvement leads to a substantial improvement in translation quality, but do not report a corresponding improvement when reordering test data in a similar fashion. Interestingly, Bisazza and Federico (2010) report that fuzzy reordering the test data improves MT output, suggesting that fuzzification may be the mechanism necessary to render reordering on test data useful. To the best of our knowledge, nobody has yet used fuzzification to correct the identified subject span of complete Arabic dependency parses. Green et al. (2009) use a conditional random field sequence classifier to detect Arabic noun phrase subjects in verb-initial clauses achieving an F-score of 61.3%. They integrate their classifier's decisions as additional features in the Moses decoder (Koehn et al., 2007), but do not show any gains.

The present work may be thought of as extending the fuzzification explored by Bisazza and Federico (2010) to the domain of full parsing—a combination, in some sense, of their approach with the work of Carpuat et al. (2010). The approach examined in this paper differs from Collins et al. (2005) in its use of fuzzification, from Bisazza and Federico (2010) in its use of a complete dependency parse, and from Carpuat et al. (2010) in its use of a reordered test set.

Figure 1: An example of a dependency tree of a Verb-Object-Subject Arabic sentence: هز الرياض مساء اليوم هجومان سيارتين مفخختين ب+ *hz AlryAD msA' Alywm hjwmAn b+ syArtyn mfxxtyn* 'Two car bombs shook Riyadh this evening'. The predicted tree (on the left) shows an incorrect subject span (words 5-8).



3 Motivation

While the VSO order is common at both the matrix and non-matrix level in Arabic newswire text, matrix VSO constructions are almost always reordered in translation, while non-matrix VSO constructions are frequently translated monotonically (they are instead passivized or otherwise transformed in a fashion that leaves them parallel to the source Arabic text) (Carpuat et al., 2010). This reordering, as noted in the introduction, is notoriously difficult for phrase-based statistical machine translation systems to capture. It is further exacerbated by the low quality of Arabic parsing especially for subject span identification (Green et al., 2009).

3.1 Reordering

We began by performing a series of reordering experiments using gold-standard parses of the NIST MT05 data set:[1] (a) a baseline experiment with no reordering, (b) an experiment which forced reordering on all matrix subjects, and (c) an experiment in which the translation system was presented with a lattice, in which one path contained the original sentence and the other path contained the sentence with the matrix subject reordered. The baseline system produced a BLEU score of 47.13, forced reordering produced a BLEU score of 47.43, and optional reordering produced a BLEU score of 47.55. These results indicate that, given correct reordering boundaries, the translation quality can indeed be improved with reordered test data. Furthermore, the improvement noted above between the forced reordering and optional reordering experiments, while small, indicates that even with correct parses it is sometimes preferable to leave the input sentence un-reordered. This is consistent with Carpuat et al. (2010)'s ob-

[1]The gold parses for NIST MT05 are part of the Columbia Arabic Treeebank (CATiB) (Habash and Roth, 2009).

servation that even VS-ordered matrix verbs in Arabic are sometimes translated monotonically into English (as, for example, in passive constructions). An alternative explanation may be that since the training data itself is not re-ordered, it is plausible that some re-ordering may cause otherwise good possible matches in the phrase table to not match any more.

3.2 Parser Error

The problem of finding correct subject span boundaries for reordering, however, is a particularly difficult one. Both Habash (2007b) and Green et al. (2009) have noted previously that even state-of-the-art Arabic dependency parsers tend to perform poorly, and we would expect that incorrect boundaries would do more harm than good for translation. In order to determine how to "fix" these spans, it is first necessary to understand the kinds of errors that the parser makes. A set of predicted parses of the NIST MT05 data was compared to the gold parses of the same data set.

There are three categories of error the parser can make in identifying subjects: labeling errors, attachment errors and span errors. In labeling errors, the parser either incorrectly marks a node `SBJ` when no such label appears in the gold tree, or fails to identify one of the gold-labeled `SBJ`s. In attachment error, the identified subject is marked as depending on the wrong node. Finally, in span error, the descendants assigned to a labeled SBJ are wrong. The distribution of parser errors in the NIST MT05 data is as follows:

- Label errors: 19.8% of predicted subjects are not gold subjects, and 19.1% of gold subjects are not identified as predicted subjects.
- Attachment errors: 16.92% of gold subjects are incorrectly attached in the predicted tree.
- Span errors: 26.4% of predicted subject spans are incorrect.

In this paper, we focus on correcting the largest sources of error: incorrect span and false-positive subjects. We now provide further analysis of the span errors.

In principle, spans can be marked incorrectly both on their front and back ends; however, because left-dependency is fairly uncommon in Arabic and happens in a limited number of predictable cases, the parser made so few errors in identifying the left boundary of spans (1.8%) that it is not worth trying to correct them.[2]

The question is thus how to correct the right edge of spans assuming that label and attachment have been predicted correctly. Span classifications can be broken into three categories: those that are too long (i.e. that have too many right descendants), too short (i.e. that have too few right descendants), or correct (so that the predicted tree has all the same descendants as the gold tree, without regard to their syntactic structure). A comparison of gold and predicted trees for MT05 was conducted, revealing the distribution shown in Table 1. We see that the 26.4% of subjects with incorrect spans are roughly equally divided between subjects that are too short and subjects that are too long.

Type	#	%
Long	260	12.4%
Short	293	14.0%
Correct	1538	73.6%
Total	2091	100%

Table 1: Distribution of span errors in NIST MT05

To gain further insight into the nature of the subject span errors, we examined more closely the 26.4% of cases where the span is incorrectly labeled, looking specifically at the "difference box": the set of contiguous nodes that must be added to or removed from the predicted span to bring it into agreement with the gold span (see Fig. 1).[3] Specifically, we wished to know how many top-level constituents required addition or removal to cover the entire difference. The smaller the number of top-level constituents that needs to be added, the fewer reordering variations possible, and the better the expected performance of the system.

Roughly 2% of these difference boxes are what we might call "pathological" cases: due to some se-

[2] A note on terminology: "left" and "right" are used throughout this paper with reference to word order when using the Latin alphabet. "Left" should be understood to mean "towards the beginning of the sentence", and "right" to mean "towards the end of the sentence."

[3] Arabic transliteration is presented in the Habash-Soudi-Buckwalter scheme (Habash et al., 2007).

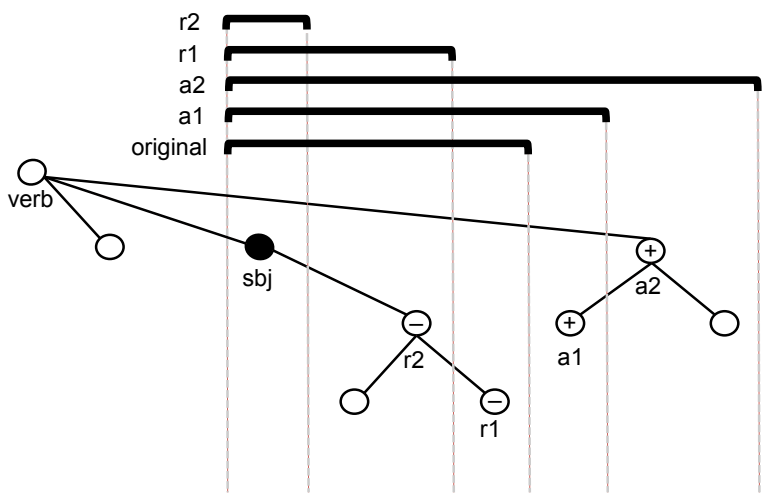


Figure 2: A schematic representation of the fuzzification algorithm. The black node is the matrix subject, + indicates that a node (and its descendants) can be added, − indicates that a node (and its descendants) can be removed, and the black brackets denote the boundaries of the candidate spans.

rious error in parsing, there is a constituent inside the difference box with descendants outside the box. These are algorithmically very difficult to correct as they require us to either add a constituent and then prune it, or remove a constituent and then reattach some of its children; attempting to correct for this possibility in all sentences will lead to a combinatorial explosion of possible parses. Fortunately, these pathological cases make up a small enough portion of the data set that they can be safely disregarded.

More promisingly, 66.5% of incorrect spans can be corrected with the addition or removal of a single constituent; in other words, the recall of span identification can be improved from 73.6% to 91.2% by adding or removing at most one constituent at the end of the parser's identified span.

4 Approach

To improve translation of matrix subjects, we implement fuzzy reordering by using a lattice-based approach similar to Bisazza and Federico (2010) to correct the matrix subject spans identified by a state-of-the-art dependency parser (Marton et al., 2010). Specifically, we take a twofold approach to fuzzy reordering. First, we present the translation system with both un-reordered and reordered options. This is motivated by the observation that on gold parses, optional reordering outperformed forced reordering (Section 3.1). Second, we apply a fuzzification algorithm to the reordered subject span, adding yet more options to the lattice. This is motivated by the observation that the greatest source of parsing errors in subjects is span errors (Section 3.2). We discuss these two techniques in turn.

4.1 Optional Reordering

In keeping with results from the initial gold experiments, we decided to generate a lattice identical to that used for the optional-reordering experiment, in which the translation system was presented with the input sentence both un-reordered and reordered, using a predicted parse to perform the reordering.

4.2 Subject Span Fuzzification

The observation that 91.2% of spans can be recalled with single-constituent modifications led very naturally to the following fuzzification algorithm, which is illustrated in Fig. 2:

1. For each matrix subject in the parse tree[4], create an empty list to hold fuzzified boundaries.
2. Original span: Add to the list the tuple (l, r, v), where l is the index of the predicted span's leftmost descendant, r is the index of the predicted span's rightmost descendant and v is the verb

[4]Allowance must be made for parsers which incorrectly identify multiple subjects for the matrix verb.

that the predicted span attaches to. (This step produces the span labeled "original" in Fig. 2.)

3. Expansion: Add to the list all tuples of the form (l, r^+, v), where r^+ is the index of the rightmost descendant of a node whose leftmost descendant has index $r + 1$. (This step produces the spans labeled "a1" and "a2" in Fig. 2.)
4. Contraction: Add to the list all tuples of the form $(l, r^- - 1, v)$, where r^- is the index of the leftmost descendant of a node whose rightmost descendant has index r. (This step produces the spans labeled "r1" and "r2" in Fig. 2.)
5. Create the list of all valid combinations of spans by taking the Cartesian product of all the per-subject span lists, and rejecting all entries in which two spans overlap. (This step accounts for multiple subject cases.)

The result of this algorithm is a list of lists of tuples, where each tuple defines a single reordering, and each list of tuples defines a set of spans that must be moved to the left of the matrix verb for one reordering. These re-orderings are then joined together to form the final lattice. If a single-constituent correction to the span exists (except in the aforementioned pathological and left-attachment cases), it is guaranteed to appear as one path through the lattice.

5 Evaluation

5.1 Experimental Setup

We used the open-source Moses PSMT toolkit (Koehn et al., 2007). Training data was a newswire (MSA-English) parallel text with 12M words on the Arabic side (LDC2007E103)[5] Sentences were reordered only for alignment, following the approach of Carpuat et al. (2010). Parses were obtained using a publicly available parser for Arabic (Marton et al., 2010). GIZA++ was used for word alignment (Och and Ney, 2003) and phrase translations of up to 10 words are extracted in the Moses phrase table. The same baseline phrase table was used in all experiments.

The system's language model was trained both on the English portion of the training corpus and English Gigaword (Graff and Cieri, 2003). We used a 5-gram language model with modified Kneser-Ney smoothing implemented using the SRILM toolkit (Stolcke, 2002). Feature weights were tuned with MERT (Och, 2003) to maximize BLEU on the NIST MT06 corpus. MERT was done only for the baseline system; these same weights were used for all experiments to control for the effect of MERT instability. In the future, we plan to experiment with approach-specific optimization and to use recent published suggestions on controlling for optimizer instability (Clark et al., 2011).

English data was tokenized using simple punctuation-based rules. Arabic data was segmented with to the Arabic Treebank tokenization scheme (Maamouri et al., 2004) using the MADA+TOKAN morphological disambiguator and tokenizer (Habash and Rambow, 2005; Habash, 2007a; Roth et al., 2008). The Arabic text was also Alif/Ya normalized (Habash, 2010). MADA-produced Arabic lemmas were used for word alignment.

We compare four settings with predicted parses (as opposed to the gold parse experiments discussed in Section 3):

- **BASE** An un-reordered test set;
- **FORCE** A test set which forced reordering on matrix verbs;
- **OPT** A test set with fuzzification through optional reordering on matrix verbs; and
- **SPAN** A test set with fuzzification through optional reordering on matrix verbs and through fuzzification of the subject span according to the algorithm shown in Section 4.2.

Each reordering corpus used Moses' lattice input format (Dyer et al., 2008) (including the baselines, which had only one path). Results are presented in terms of the standard BLEU metric (Papineni et al., 2002), METEOR metric (Banerjee and Lavie, 2005) and a manual evaluation targeting subject span translation correctness.

5.2 Automatic Evaluation Results

Table 2 presents the results for the experiments discussed above. Columns three and Four (Prec-1g and Prec-4g) indicate the corresponding 1-gram and 4-gram (sub-BLEU) precision scores, respectively.

[5] All data is available from the Linguistic Data Consortium: http://www.ldc.upenn.edu.

System	BLEU	Prec-1g	Prec-4g	METEOR
BASE	47.13	81.91	29.52	53.09
FORCE	47.03	81.78	29.52	53.11
OPT	47.42	81.88	30.04	53.22
SPAN	47.41	81.92	30.03	53.21

Table 2: Automatic evaluation results

Both OPT and SPAN showed a statistically significant improvement in BLEU score over BASE and FORCE above the 95% level. Statistical significance is computed using paired bootstrap resampling (Koehn, 2004). The difference between OPT and SPAN, however, was not statistically significant.

The relatively small difference in BLEU score between the baseline and *gold* reordering (Section 3: baseline 47.13 and optional reordering 47.55) suggests that we should expect at most a modest increase in BLEU from improving the predicted trees.

The first key observation in these results is that with a noisy parser, translation quality actually goes down with forced reordering—the opposite of what was observed in the gold experiment. By introducing either optional reordering or complete fuzzification, however, BLEU score increases .3 past the baseline to achieve nearly three quarters of the gain obtained by optional reordering using the gold parse (Section 3: baseline 47.13 and optional reordering 47.55). In other words, it is possible to compensate for the parser noisiness without actually attempting to correct spans: simply allowing the translation system to fall back on an un-reordered input leads to a significant gain in BLEU.

One possible explanation for this fact is that we only ever correct for parses on the right-hand side—the left sides are virtually always correct. Thus, when we perform any reordering, even if the subject span is not entirely perfect, we guarantee that we bring at least one word from the sentence (and usually more) into alignment where it was out of alignment before; this obviously leads to better BLEU n-gram scores along that boundary.

The general trend in these results is confirmed by the results of a METEOR analysis, also provided in Tab. 2. Again, both the OPT and SPAN systems result exhibit comparable performance, and demonstrate an improvement over the baseline.

The second observation is that introducing span fuzzification did not improve over simple optional reordering. There are a several reasons this could be happening:

- The increased fluency and introduction of unseen phrases cancel each other out.
- All the gains that come from reordering occur at the left; the presence or absence of correct words at the right end is less important.
- Better sentences are proposed during the translation process, but they are not selected during the final filtering stage.
- The sentences being output are actually better, but the improvement is not captured by the automatic evaluation.

Further experiments will be necessary to determine whether any of the first three possibilities is the case. We next consider the fourth possibility in more detail.

5.3 Manual Evaluation

We additionally conducted a manual evaluation to examine how subject quality differed in fuzzified vs. unfuzzified parses. Each sentence examined was assigned one of the six labels below. Examples are with respect to the reference sentence *"Recep Tayyip Erdogan announced that Turkey is strong."*

- **MM**: both verb and subject missing. *"Turkey is strong."*
- **MV**: verb missing. *"Recep Tayyip Erdogan Turkey is strong."*
- **MS**: subject missing. *"announced that Turkey is strong."*
- **SO**: subject overlaps with verb. *"Recep announced Tayyip Erdogan Turkey is strong."*
- **SI**: verb precedes subject (as in Arabic). *"announced Recep Tayyip Erdogan that Turkey is strong."*
- **C**: verb follows subject (as in English), i.e. the correct ordering. *"Recep Tayyip Erdogan announced that Turkey is strong."* We also include in this category sentences where the English reference contains no verb (e.g. in newspaper headlines).

System	MM	MS	MV	SI	SO	C	M*	S*	C
BASE	8	13	11	9	3	53	33	12	53
OPT	7	11	10	5	5	61	28	10	61
SPAN	8	10	09	5	2	64	27	7	64

Table 3: Subject integrity analysis results. All numbers are %s.

By grouping some of these categories together, we obtained the following label scheme:

- **M***: MM, MV or MS, i.e. verb or subject is missing.
- **S***: SO or SI, i.e. word order is incorrect.
- **C**: as above.

280 sentences selected randomly from our test set were evaluated, generating 461 unique output sentences. Annotation was performed by two English speakers, with 40 input sentences (68 unique outputs) annotated by both authors to collect agreement statistics. For the complete label scheme, the annotators agreed on 86.8% of labels, with Cohen's $\kappa = 0.811$. For the simple label scheme, the annotators agreed on 92.6% of labels, with $\kappa = .883$. Results for the BASE, OPT and SPAN systems are shown in Table 3. Each annotator's labels were assigned a weight of .5 in the section that was jointly annotated.

Again, both the OPT and SPAN systems display statistically significant improvements over the baseline system ($p < 0.001$). While the SPAN system consistently displays better results than the OPT system, the significance is low ($p < .3$). Statistical significance was measured using the McNemar test of statistical significance (McNemar, 1947).

These results thus agree with the BLEU score in indicating that the OPT and SPAN systems are substantially better than the baseline, but statistically indistinguishable from each other. They further indicate that most of the improvements in the OPT system come from preventing dropped subjects or verbs, while the improvements in the SPAN system result in roughly equal proportion from preventing word-dropping and ensuring correct ordering.

6 Conclusion & Future Work

We presented an approach for improving Arabic-English PSMT using syntactic information from a noisy parser. We demonstrated that translation quality goes down with forced reordering, but improves with the introduction of either optional reordering and subject span fuzzification. The BLEU score increases by 0.3% absolute past the baseline achieve nearly three quarters of the maximum possible gain starting with gold parses. A detailed manual evaluation produces results generally consistent with BLEU, but highlights the small improvements that can be gained by subject span fuzzification.

In the future, we plan to explore a more sophisticated approach to the lattice of re-orderings presented here. We would take into account the fact that it is possible to suggest to the system that certain re-orderings are less likely than others without removing them from the search space completely. The same can be done for the fuzzification task: while we might wish to add additional fuzzification options, we also don't want the correct choice to be crowded out by too many alternatives.

Acknowledgements

We would like to thank Marine Carpuat, Yuval Marton, Amit Abbi and Ahmed El Kholy for helpful discussions and feedback. The second and third authors were supported by the Defense Advanced Research Projects Agency (DARPA) under GALE Contract No HR0011-08-C-0110. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA.

References

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.

Arianna Bisazza and Marcello Federico. 2010. Chunk-based verb reordering in VSO sentences for Arabic-English statistical machine translation. In *Proceedings of ACL 2010: Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, Uppsala, Sweden.

Marine Carpuat, Yuval Marton, and Nizar Habash. 2010. Improving Arabic-to-English Statistical Machine Translation by Reordering Post-Verbal Subjects for Alignment. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 178–183, Uppsala, Sweden, July.

Jonathan H. Clark, Chris Dyer, Alon Lavie, and Noah A. Smith. 2011. Better hypothesis testing for statistical machine translation: Controlling for optimizer instability. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 176–181, Portland, Oregon, USA.

Michael Collins, Philipp Koehn, and Ivona Kucerova. 2005. Clause Restructuring for Statistical Machine Translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 531–540, Ann Arbor, Michigan.

Chris Dyer and Philip Resnik. 2010. Context-free reordering, finite-state translation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 858–866, Los Angeles, California.

Christopher Dyer, Smaranda Muresan, and Philip Resnik. 2008. Generalizing word lattice translation. In *Proceedings of ACL-08: HLT*, Columbus, Ohio.

Jakob Elming and Nizar Habash. 2009. Syntactic Reordering for English-Arabic Phrase-Based Machine Translation. In *Proceedings of the EACL 2009 Workshop on Computational Approaches to Semitic Languages*, pages 69–77, Athens, Greece, March.

J. Elming. 2008. Syntactic reordering integrated with phrase-based smt. In *Proceedings of the ACL Workshop on Syntax and Structure in Statistical Translation (SSST-2)*.

David Graff and Christopher Cieri. 2003. English Gigaword, LDC Catalog No.: LDC2003T05. Linguistic Data Consortium, University of Pennsylvania.

Spence Green, Conal Sathi, and Christopher D. Manning. 2009. NP Subject Detection in Verb-initial Arabic Clauses. In *Proceedings of the Third Workshop on Computational Approaches to Arabic Script-based Languages (CAASL3)*.

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 Ryan Roth. 2009. CATiB: The Columbia Arabic Treebank. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 221–224, Suntec, Singapore.

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. 2007a. 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. 2007b. Syntactic preprocessing for statistical machine translation. In *Proceedings of the 11th MT Summit*.

Nizar Habash. 2010. *Introduction to Arabic Natural Language Processing*. Morgan & Claypool Publishers.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Christopher Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Christopher Dyer, Ondrej Bojar, Alexandra Constantin, and Evan 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 formachine translation evaluation. In *Proceedings of the Empirical Methods in Natural Language Processing Conference (EMNLP'04)*, Barcelona, Spain.

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.

Yuval Marton, Nizar Habash, and Owen Rambow. 2010. Improving Arabic Dependency Parsing with Lexical and Inflectional Morphological Features. In *Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 13–21, Los Angeles, CA, USA, June.

Quinn McNemar. 1947. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157.

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.

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.

Andreas Stolcke. 2002. SRILM - an Extensible Language Modeling Toolkit. In *Proceedings of the International Conference on Spoken Language Processing (ICSLP)*, volume 2, pages 901–904, Denver, CO.

Fei Xia and Michael McCord. 2004. Improving a statistical mt system with automatically learned rewrite patterns. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING 2004)*, pages 508–514, Geneva, Switzerland.