Mach Translat (2012) 26:137–157
DOI 10.1007/s10590-011-9111-z

Soft syntactic constraints for Arabic–English hierarchical phrase-based translation

Yuval Marton · David Chiang · Philip Resnik

Received: 3 July 2010 / Accepted: 22 August 2011 / Published online: 26 October 2011

Abstract In adding syntax to statistical machine translation, there is a tradeoff between taking advantage of linguistic analysis and allowing the model to exploit parallel training data with no linguistic analysis: translation quality versus coverage. A number of previous efforts have tackled this tradeoff by starting with a commitment to linguistically motivated analyses and then finding appropriate ways to soften that commitment. We present an approach that explores the tradeoff from the other direction, starting with a translation model learned directly from aligned parallel text, and then adding soft constituent-level constraints based on parses of the source language. We argue that in order for these constraints to improve translation, they must be fine-grained: the constraints should vary by constituent type, and by the type of match or mismatch with the parse. We also use a different feature weight optimization technique, capable of handling large amount of features, thus eliminating the bottleneck of

Yuval Marton was at University of Maryland, College Park, at the time the experiments described here took place.

Y. Marton (✉)
IBM T.J. Watson Research Center, 1101 Kitchawan Road / Route 134,
Yorktown Heights, NY 10598, USA
e-mail: yymarton@us.ibm.com

D. Chiang
USC Information Sciences Institute (ISI), 4676 Admiralty Way, Suite 1001,
Marina del Rey, CA 90292, USA
e-mail: chiang@isi.edu

P. Resnik
Department of Linguistics and the Laboratory for Computational Linguistics and
Information Processing (CLIP) at the Institute for Advanced Computer Studies (UMIACS),
University of Maryland, College Park, MD 20742-7505, USA
e-mail: resnik@umiacs.umd.edu

feature selection. We obtain substantial improvements in performance for translation from Arabic to English.

Keywords Machine translation · Syntax · Soft constraints · Arabic · Parsing · Statistical methods

1 Introduction

In adding syntax to statistical machine translation (SMT), there is a tradeoff between taking advantage of linguistic analysis and allowing the model to exploit mappings learned from data without linguistic analysis (except perhaps word segmentation). This is a tradeoff of translation quality versus coverage, analogous to precision versus recall. This work combines these two knowledge sources, starting with a synchronous context-free grammar-based translation model learned directly from aligned parallel text, and then adding soft constituent-level constraints based on syntactic parses of the source language. We argue that in order for these constraints to improve translation, they must be fine-grained. This article draws together work by Marton and Resnik (2008) and subsequent work by Chiang et al. (2008), revising and extending these previous publications with analysis based on Chapter 2 of the first author's doctoral dissertation (Marton 2009).

The statistical revolution in machine translation (Brown et al. 1990, 1993) replaced an earlier paradigm of detailed language analysis with automatic learning of shallow source-target mappings from large parallel corpora. Over the last several years, however, the pendulum has begun to swing back in the other direction, with researchers exploring a variety of statistical models that take advantage of syntactic analysis, especially in the target language (e.g., Cowan et al. (2006); Zollmann and Venugopal (2006); Marcu et al. (2006); Galley et al. (2006); Cherry (2008); Mi et al. (2008); Xiong et al. (2009); Venugopal et al. (2009); DeNeefe and Knight (2009) and numerous others).

Chiang (2005) distinguishes statistical machine translation approaches that are syntactic in a *formal* sense from those that are syntactic in a *linguistic* sense: Formally syntactic approaches go beyond the finite-state underpinnings of phrase-based models, using grammars such as synchronous context-free grammar (SCFG). Linguistically syntactic approaches take advantage of a priori language knowledge in the form of annotations derived from human linguistic analysis or treebanking. The two forms of syntactic modeling are doubly dissociable: current research frameworks include systems that are finite state but informed by linguistic annotation prior to training (e.g., Koehn and Hoang (2007); Birch et al. (2007); Hassan et al. (2007)), and also include systems employing context-free models trained on parallel text without benefit of any prior linguistic analysis (e.g. Chiang (2005, 2007); Wu (1997)). Over time, however, there has been increasing movement in the direction of systems that are syntactic in both the formal and linguistic senses (see Table 1).

Table 1 Formally and linguistically syntactic SMT approaches are doubly dissociable

	Data-driven	Linguistically syntactic
Word-based or flat phrase-based	IBM models (Brown et al. 1993), Pharaoh (Koehn 2004), Moses (Koehn et al. 2007)	Koehn and Hoang (2007), Birch et al. (2007), Hassan et al. (2007), Cherry (2008),…
Hierarchical, formally syntactic	ITG (Wu 1997), SCFG: Hiero (Chiang 2005, 2007),…	Cowan et al. (2006), Zollmann and Venugopal (2006), Marcu et al. (2006), Galley et al. (2006), Marton and Resnik (2008), Chiang et al. (2008), Xiong et al. (2009), DeNeefe and Knight (2009),…

In any such system, there is a natural tension between taking advantage of linguistic analysis versus allowing the model to use mappings learned from non-annotated parallel training data. The tradeoff often involves starting with a system that exploits rich linguistic representations and relaxing some part of it. For example, Zollmann and Venugopal (2006) begin with a string-to-tree model using treebank-based target language analysis, and relax the notion of syntactic constituency in a manner similar to categorial grammar, in order to admit more translation rules.

Here we address this challenge from a less explored direction. Rather than starting with a system based on linguistically motivated parse trees, we begin with a model that is syntactic only in the formal sense. We then introduce soft constraints that take source-language parses into account to a limited extent. Introducing syntactic constraints in this restricted way allows us to take maximal advantage of what can be learned from parallel training data, while effectively factoring in key aspects of linguistically motivated analysis. As a result, we obtain substantial improvements in performance for Arabic–English translation.

We build on the Hiero SMT framework (Chiang 2005, 2007), briefly reviewed in Section 2, including Chiang's initial effort to incorporate soft source-language constituency constraints for Chinese-English translation. We suggest that an insufficiently fine-grained view of constituency constraints was responsible for Chiang's lack of strong results. Our contribution is to introduce finer-grained constraints into the model, and a novel type of syntactic constraint, penalizing source-side translation units that cross the boundaries of syntactic constituents (Section 3). We carry out experiments on Arabic–English translation (Section 4), and show gains when optimizing the models using the standard Minimum Error Rate Training (MERT) weight optimization algorithm, and also when using the more recent Margin Infused Relaxed Algorithm (MIRA), one of whose advantages is handling a large amount of features. We then discuss the results (Section 5), review related work (Section 6), and conclude with a summary and potential directions for future work (Section 7).

2 Background

2.1 Hierarchical phrase-based translation (Hiero)

Hiero (Chiang 2005, 2007), which is used in the experiments reported here, is a hierarchical phrase-based SMT framework that generalizes phrase-based models by permitting phrases with gaps. Formally, Hiero's translation model is a weighted synchronous context-free grammar (SCFG). Hiero employs a generalization of the standard non-hierarchical phrase extraction approach in order to acquire the synchronous rules of the grammar directly from word-aligned parallel text. Rules have the form $\mathrm{X} \rightarrow \langle \bar{e}, \bar{f} \rangle$ where $\bar{e}$ and $\bar{f}$ are phrases containing terminal symbols (words) and possibly co-indexed instances of the nonterminal symbol X.[1] For example, the translation rule

$$\mathrm{X} \rightarrow \langle \textit{the green } \mathrm{X}_1 \textit{ sleeps } \mathrm{X}_2, \textit{ la } \mathrm{X}_1 \textit{ verte dort } \mathrm{X}_2 \rangle$$

could translate the English *"the green caterpillar sleeps under a leaf"* to the French *"la chenille verte dort sous une feuille"*, or translate the English *"the green idea sleeps furiously"* to the French *"la idée (l'idée) verte dort furieusement"*. All co-indexed occurrences of X would have to be translated with another such rule, e.g., $\mathrm{X} \rightarrow \langle \textit{idea}, \textit{idée} \rangle$ or $\mathrm{X} \rightarrow \langle \textit{furiously}, \textit{furieusement} \rangle$. The English (source) side of the nested rule will substitute a source side occurrence of X in the containing rule, while the target side of the nested rule will synchronously substitute the occurrence of X in the containing rule which was co-indexed with the substituted source side X. Since Hiero is SCFG-based, the choice of what nested rule to use is independent of the containing rule.

Associated with each rule is a set of translation model features, $\phi_i(\bar{e}, \bar{f})$; for example, one intuitively natural feature of a rule is the phrase translation log-probability $\phi(\bar{e}, \bar{f}) = \log p(\bar{e} \mid \bar{f})$, directly analogous to the corresponding feature in non-hierarchical phrase-based models like Pharaoh (Koehn et al. 2003). In addition to this phrase translation probability feature, Hiero's feature set includes the inverse phrase translation probability $\log p(\bar{f} \mid \bar{e})$, lexical weights $\mathrm{lexwt}(\bar{f} \mid \bar{e})$ and $\mathrm{lexwt}(\bar{e} \mid \bar{f})$, which are estimates of translation quality based on word-level correspondences (Koehn et al. 2003), and a rule penalty allowing the model to learn a preference for longer or shorter derivations; see Chiang (2007) for details.

These features are combined using a linear model, with each rule contributing

$$\sum_i \lambda_i \phi_i(\bar{e}, \bar{f}) \tag{1}$$

to the total score of a derived hypothesis. Each λ_i is a weight associated with feature ϕ_i, and these weights are typically optimized using minimum error rate training

[1] This is slightly simplified: Chiang's original formulation of Hiero has two nonterminal symbols, X and S. The latter is used only in two special "glue" rules that permit complete trees to be constructed via concatenation of subtrees when there is no better way to combine them.

(Och 2003). As noted in Section 1, Hiero is only formally syntax-based, in that it assigns a hierarchical structure to a sentence, but it is not linguistically syntax-based, in that it has no knowledge of verb phrases or other syntactic constituents. Next, we discuss past and present attempts to make Hiero syntactically aware in the linguistic sense as well.

2.2 Soft syntactic constraints

When looking at Hiero rules, which are acquired automatically from parallel text, it is easy to find many cases that seem to respect linguistically motivated boundaries. For example,

$$\mathrm{X} \rightarrow \langle \text{jinnian } \mathrm{X}_1, \mathrm{X}_1 \text{ this year} \rangle,$$

seems to capture the use of *"jinnian"*/*"this year"* as a temporal modifier when building linguistic constituents such as noun phrases (*"the election this year"*) or verb phrases (*"voted in the primary this year"*). However, observe that nothing in the Hiero framework actually *requires* nonterminal symbols to cover linguistically sensible constituents, and in practice they frequently do not. This rule could just as well be applied with X_1 covering the phrase *"submitted and"* to produce the non-constituent substring *"submitted and this year"* in a hypothesis like *"The budget was submitted and this year cuts are likely"*.

Chiang (2005) conjectured that there might be value in allowing the Hiero model to favor hypotheses whose derivation respects linguistically motivated source-language constituency boundaries, as identified using a parser. He tested this conjecture by adding a soft constraint in the form of a *constituency feature*: if a rule $\mathrm{X} \rightarrow \langle \bar{e}, \bar{f} \rangle$ is used in a derivation, and the span of $\bar{f}$ is a constituent in the source-language parse, then a term λ_c is added to the model score in expression (1).[2] A hard constraint would prevent the application of any rule violating syntactic boundaries, whereas a *soft* constraint (implemented as a weighted feature) merely encourages rule applications matching syntactic constituent boundaries in the source language by boosting the associated scores. The weight λ_c, like all the other λ_i, is set during a tuning step, which determines empirically the extent to which the constituency feature should be trusted.

Figure 1 illustrates the way the constituency feature (denoted here all-labels$^{=}$) worked, treating English as the source language for readability. In this example, the reward λ_c would be added to the hypothesis score for any rule used in the hypothesis whose source side covers one of the spans indicated by the top five horizontal bars: *"the minister"*, *"a speech"*, *"yesterday"*, *"gave a speech yesterday"*, or *"the minister gave a speech yesterday."* A rule translating, say, *"minister gave a"* (bottom horizontal bar) would receive no reward.

[2] Formally, $\phi_c(\bar{e}, \bar{f})$ is defined as a binary feature, with value 1 if $\bar{f}$ spans a source constituent and 0 otherwise. In the latter case $\lambda_c \phi_c(\bar{e}, \bar{f}) = 0$ and the score in expression (1) is unaffected.

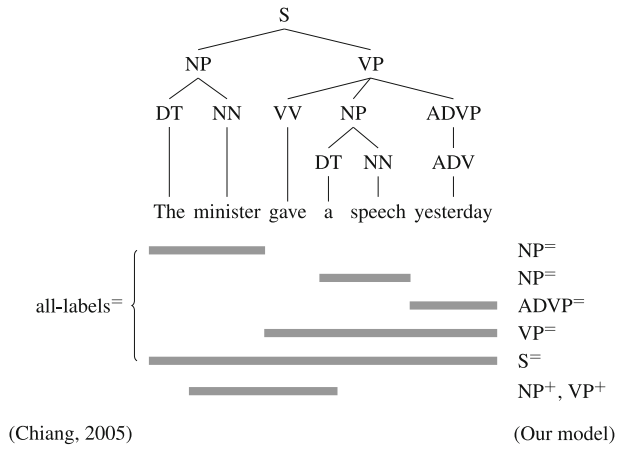


Fig. 1 Translation rules whose source side exactly spans any constituent type (top five horizontal bars) are equally rewarded by Chiang's feature all-labels$^{=}$, while a rule translating *"minister gave a"* (*bottom bar*) is not rewarded. Our models give *separately weighted* rewards for each constituent type (top five bars) and penalties for crossing each constituent type (*bottom bar*). This example is in English for ease of readability

Chiang tested the constituency feature for Chinese-English translation, and obtained no significant improvement on the test set. The constituency feature was not pursued again in a later discussion (Chiang 2007).

3 Soft syntactic constraints, revisited

Why did the soft constraints in Chiang (2005) yield negative results? On the face of it, there are any number of possible reasons, including practical issues like the quality of the Chinese parses (although it turned out not to be the issue, as we see in Section 4). We focus here on two conceptual issues underlying the use of source language syntactic parses there.

First, the constituency feature treats all syntactic constituent types equally, making no distinction among them. In Fig. 1, the same reward is applied to all of the spans indicated by the top five horizontal bars, even though they have different syntactic categories. For any given language pair, however, there might be some source constituents that tend to map to the target language as units more naturally than others (Fox 2002; Eisner 2003; Koehn 2003). Moreover, a parser may tend to be more accurate for some constituents than for others. Assigning a high weight to noisy parsing tags or to inconsistent source-target constituent pairing may cause more damage than benefit to the overall translation quality.

Second, the Chiang (2005) constituency feature gives a rule additional credit only when the rule's source side overlaps exactly with a source-side syntactic constituent. Logically, however, it might make sense not just to give a rule $X \rightarrow \langle \bar{e}, \bar{f} \rangle$ extra credit when $\bar{f}$ matches a constituent, but also to incur a *penalty* when $\bar{f}$ *violates* a constituent boundary. For example, in Fig. 1, one might want to penalize hypotheses containing

rules whose $\bar{f}$ is *"minister gave a"* (or *"minister gave"*, *"the minister gave a"*, and so forth).

These observations suggest a finer-grained approach to the constituency feature idea, retaining the idea of soft constraints, but applying them using *various* soft-constraint constituency features. Our first observation argues for distinguishing among constituent types (NP, VP, etc.). Our second observation argues for distinguishing the benefit of matching constituents from the cost of crossing constituent boundaries. We therefore define a set of new features: for each nonterminal symbol X $\in$ {SBAR, S, NP, VP, . . .},

- $X^{=}$ fires when $\bar{f}$ *matches* a constituent of type X in the source-language parse
- X^{+} fires when $\bar{f}$ *crosses* a constituent of type X in the source-language parse

For example, $\phi_{\text{NP}=}$ would denote a binary feature that fires whenever the span of a rule's source side $\bar{f}$ exactly matches an NP in the source-side parse tree, resulting in $\lambda_{\text{NP}=}$ being added to the hypothesis score (equation (1)). In Fig. 1, this feature would reward rules with spans indicated by the top two horizontal bars, but not the other bars. Similarly, $\phi_{\text{VP}+}$ would denote a binary feature that fires whenever the span of $\bar{f}$ crosses a VP boundary in the parse tree, resulting in $\lambda_{\text{VP}+}$ being added to the hypothesis score. [3] In Fig. 1, this feature would fire on the span indicated by the bottom bar. For readability from this point forward, we will omit ϕ from the notation and refer to features such as $\text{NP}^{=}$ (which one could read as "NP match"), VP^{+} (which one could read as "VP crossing"), etc.

For completeness, one might want a set of features that characterizes all possible relationships that $\bar{f}$ can have to the input parse tree. Such a set might include not only

- $\bar{f}$ matching a constituent exactly ($X^{=}$)
- $\bar{f}$ crossing a constituent boundaries (X^{+})

but also

- $\bar{f}$ being properly contained within the constituent span (e.g., in Fig. 1, the second bar *"a speech"* for the VP *"gave a speech yesterday"*),
- $\bar{f}$ properly containing the constituent span (e.g., the fourth bar *"gave a speech yesterday"* for the NP *"a speech"*), or
- $\bar{f}$ being outside the constituent span entirely (e.g., the fourth bar *"gave a speech yesterday"* for the NP *"the minister"*).

Often, when one of these latter three possibilities occur, $\bar{f}$ will exactly match or cross the boundaries of some other constituent, and therefore, a feature of the first two kinds will fire. For example, in Fig. 1 again, *"a speech"* (second horizontal bar) is properly contained in the span of the VP *"gave a speech yesterday"*, but is also a NP; therefore, although neither $\text{VP}^{=}$ nor VP^{+} would fire on a rule with this span, $\text{NP}^{=}$ would. The span *"gave a speech yesterday"* (fourth bar) properly contains the NP *"a speech"*, but is also a VP; although neither $\text{NP}^{=}$ nor NP^{+} would fire on a rule with this span, $\text{VP}^{=}$ would. Finally, *"the minister"* (first bar) is entirely outside the span

[3] This binary feature always fires with a positive value, but discriminative training should, and generally does, assign $\lambda_{\text{VP}+}$ a negative value, resulting in a penalty associated with this feature.

of the VP, but is a NP; neither $VP^{=}$ nor VP^{+} would fire on a rule with this span, but $NP^{=}$ would.

If the parse trees are binary-branching, then our existing feature set is complete in the sense that any span of $\bar{f}$ will cause one of our features to fire (it must either be a constituent or cross a constituent). But for trees with higher maximal fan-out, spans such as *"gave a speech"* and *"a speech yesterday"* in Fig. 1 are neither rewarded nor penalized by our features. We could handle such spans by binarizing the trees using simple heuristics; alternatively, we could add more feature types. For example, there could be a feature that fires when a rule's source side matches only the left or right boundary of a node of type X, or several children of a node of type X. Some of these possibilities are explored in work subsequent to ours (Xiong et al. 2009).

In addition to the individual features defined above, we define the following variants:

- For each constituent type, e.g. NP, we define a *conflated feature* NP_ that ties the weights of $NP^{=}$ and NP^{+}. If $NP^{=}$ matches a rule, the model score is incremented by $\lambda_{NP\_}$, and if NP^{+} matches, the model score is decremented by the same quantity.
- For each constituent type, e.g. NP, we define a version of the model, NP_2, in which $NP^{=}$ and NP^{+} are *both* included as features, with separate weights $\lambda_{NP^{=}}$ and $\lambda_{NP^{+}}$.
- The following nonterminal labels were selected based on their frequency in the tuning data, whether they frequently cover a span of more than one word, and whether they represent linguistically relevant constituents: SBAR, S, NP, VP, PP, ADJP, ADVP, and QP. Let XP be this set of nonterminals, and define feature $XP^{=}$ as the disjunction of $\{X^{=} \mid X \in \text{XP}\}$, i.e., its value equals 1 for a rule if the span of $\bar{f}$ exactly covers a constituent having any of the standard labels in XP. The definitions of XP^{+}, XP_, and XP_2 are analogous.
- Similarly, since Chiang's original constituency feature can be viewed as a disjunctive all-labels$^{=}$ feature, we also defined all-labels^{+}, all-labels$_2$, and all-labels_ analogously.

4 Experiments

We describe below two sets of experiments with soft syntactic constraints as weighted features in a linear model. Section 4.1 describes experiments optimizing the feature weights with the *de facto* standard minimum error rate training (MERT). Section 4.2 addresses the feature selection problem that arises in Section 4.1 using another weight optimization method, and revalidates our approach with another tuning set and a much larger training set.

4.1 MERT experiments

We carried out experiments for translation from Arabic to English,[4] using the Hiero system (Chiang et al. 2005). Language models were built using the SRI Language

[4] We refer the reader to Marton and Resnik (2008) for details of related Chinese to English experiments.

Table 2 Training corpora for Arabic–English translation

LDC ID	Description
LDC2004T17	Ar News Trans Txt Pt 1
LDC2004T18	Ar/En Par News Pt 1
LDC2005E46	Ar/En Treebank En Translation
LDC2004E72	eTIRR Ar/En News Txt

Table 3 Training, tuning, and test set sizes for Arabic–English translation

Use	Set	Size (sentences)
Training	See Table 2	100,000
Tuning	NIST MT02	663
Test	NIST MT03	1,357
Test	NIST MT06 (NIST part)	1,797
Test	NIST MT08	1,357

Modeling Toolkit (Stolcke 2002) with modified Kneser-Ney smoothing (Chen and Goodman 1998). Word-level alignments were obtained using GIZA++ (Och and Ney 2000). The baseline model used the feature set described in Section 2.

In order to compute syntactic features, we analyzed source sentences using the state of the art, treebank-trained Stanford parser v.2007-08-19 (Klein and Manning 2003a,b). In addition to the baseline condition, we added experimental conditions of the baseline model augmented with the original constituency feature (Chiang 2005), or with one of the other features as described in Section 3.

All models were optimized and tested using the NIST definition of the BLEU metric (Papineni et al. 2002) (*shortest* effective reference length), on lowercased, tokenized outputs and references. Statistical significance of difference from the baseline BLEU score was measured by using paired bootstrap re-sampling (Koehn 2004), with a sample size of 2000 pairs. Statistical significance was determined in case the 95% confidence interval (CI) of the systems' BLEU score difference did not include zero. For conciseness, this is denoted as $p < .05$ below. Similarly, a 99% CI is denoted as $p < .01$. The word "significant" is used below as a shorthand for "statistically significant" (at $p < .05$ unless specified otherwise).

We used the training corpora in Table 2, approximately 100,000 sentence pairs after GIZA++ length-ratio filtering. We trained a trigram language model using the English side of this training set, plus the English Gigaword v2 AFP and Gigaword v1 Xinhua corpora. Weight tuning with minimum error rate training was done using the NIST MT02 set. Details are given in Table 3.

Table 4 presents the results. We first tested on the NIST MT03 and MT06 (nist-text) sets. On MT03, the original, undifferentiated constituency feature did not improve over the baseline. Two individual finer-grained features (PP^{+} and $ADVP^{=}$) yielded statistically significant gains up to .4 BLEU points, and feature combinations AP_2, XP_2 and all-labels$_2$ yielded significant gains up to 1.0 BLEU. XP_2 and all-labels$_2$

Table 4 MERT Arabic–English results, sorted by MT06 BLEU score

Arabic	MT03	MT06	MT08
Baselines			
Baseline	48.0	35.7	35.7
all-labels$^{=}$ (Chiang 2005)	47.9	**36.8****	**36.8****
Single features			
VP^{+}	48.0	34.8	
AP^{+}	48.6	35.0	
S^{+}	48.2	35.2	
SBAR$^{=}$	48.2	35.2	
NP$^{=}$	48.5	35.4	
NP^{+}	48.0	35.5	
AP$^{=}$	48.0	35.7	
ADVP^{+}	48.5	35.7	
SBAR^{+}	47.6	35.8	
S$^{=}$	48.1	**36.4****	**36.5****
PP$^{=}$	48.0	**36.5****	**36.6****
VP$^{=}$	48.0	**36.6****	**36.9****
PP^{+}	**48.4****	**37.1****	**37.0****
ADVP$^{=}$	**48.2****	**37.1****	**37.2****
Feature combination			
XP^{+}	47.7	35.2	
all-labels$_2$	**49.0****$^{+}$	35.4	35.7
all-labels_	48.3	35.5	
VP$_2$	48.3	35.5	
NP$_2$	48.3	35.6	
ADVP.VP.PP.S$^{=}$	48.3	35.7	
VP_	48.3	36.0	
all-labels^{+}	48.3	36.0	
XP$_2$	**48.6****$^{+}$	36.1	**36.1****
S$_2$	47.9	**36.1***	35.9
S_	47.9	**36.4***	**36.5****
XP$^{=}$	48.1	**36.6****	**37.0****$^{+}$
VP$^{=}$**.PP**$^{+}$**.ADVP**$^{=}$	**48.3****	**36.8****	**37.2****
AP$_2$	**48.4****	**36.9****	**37.2****
PP^{+}**.ADVP**$^{=}$	47.8	**37.1****	**36.8****
ADVP$_2$	48.0	**37.7****$^{++}$	**37.4****$^{+}$

Boldface scores indicate statistical significance: *,**: Better than baseline ($p < .05$, $p < .01$, respectively). $^{+}$,$^{++}$: Better than all-labels$^{=}$ ($p < .05$, $p < .01$, respectively). The dot in conditions such as PP^{+}.ADVP$^{=}$ denotes a model combining several features, here both PP^{+} and ADVP$^{=}$

also improved significantly on the undifferentiated constituency feature, by .7 and 1.1 BLEU, respectively.

For MT06, Chiang (2005) all-labels$^{=}$ feature improved over the baseline significantly; this is a new result, since Chiang (2005) did not experiment with Arabic. Our individual features $S^{=}$, $PP^{=}$, and $VP^{=}$ also improved over the baseline significantly, with PP^{+} and $ADVP^{=}$ achieving highest individual improvements up to 1.4 BLEU over the baseline.

More importantly, several conditions combining features achieved statistically significant gains up to almost 2 BLEU points: XP_2, S_2, S, $VP^{=}.PP^{+}.ADVP^{=}$, AP_2, $PP^{+}.ADVP^{=}$, and $ADVP_2$. (The dot in conditions such as $PP^{+}.ADVP^{=}$ denotes a model augmented with several features, here both PP^{+} and $ADVP^{=}$). Of these, $ADVP_2$ is also a significant gain over the undifferentiated constituency feature all-labels$^{=}$ ($p < .01$). We tested the best-performing models on a new test set, NIST MT08. Similar patterns reappeared: gains up to 1.7 BLEU ($p < .01$) over the baseline, with $ADVP_2$ again in the lead, also outperforming all-labels$^{=}$ ($p < .05$).

4.2 MIRA experiments

One major weakness of the experiments described in Section 4.1 was the need for feature selection: no single constituent-sensitive feature, single constraint type (matching or crossing syntactic constituent boundaries), or single combination performed the best in all test sets. Moreover, feature combination often resulted in a performance drop. Feature selection was necessary because the commonly used MERT algorithm (Och 2003) performs poorly when attempting to optimize weights of more than 20–25 features, in the experience of many researchers and our own. This section sidesteps the feature selection problem by using the Margin-Infused Relaxed Algorithm.[5] Soft syntactic constraint features, similar to those described in Section 4.1, are tested in an Arabic–English translation task. Unlike in Section 4.1, here MIRA makes it possible to tune all syntactic features in a single model. It is also worth noting that this experimentation is on a considerably larger scale than what is described in Section 4.1 and Marton and Resnik (2008)—and yet our approach shows gains here as well.

The Margin Infused Relaxed Algorithm (MIRA) is a large-margin training algorithm for structured classification, similar in spirit to more familiar large-margin methods like support vector machines. Moreover, MIRA is an online algorithm, updating model weights training example by training example. The convexity of the objective function combined with online training result in dramatically improved scalability as compared with MERT. See Crammer and Singer (2003); Watanabe et al. (2007); Chiang et al. (2008, 2009) for detailed discussion.

The baseline model was Hiero with the following baseline features: 1. two language models; 2. phrase translation probabilities $p(f|e)$ and $p(e|f)$; 3. lexical weights in both directions (Koehn et al. 2003); 4. penalties for: (a) length (word penalty); (b)

[5] This section mainly draws on Chiang et al. (2008), which includes Chiang's re-implementation of the features described in Marton and Resnik (2008), and the implementation of MIRA. Chiang et al. (2008) also introduce *structural distortion* features, which are not covered in this article.

Table 5 Training corpora for Arabic–English translation (MIRA)

LDC ID	Description
LDC2004T17[a]	Arabic News Translation Text Part 1
LDC2004T18	Arabic English Parallel News Part 1
LDC2005E46[a]	Arabic Treebank English Translation
LDC2004E13	UN Arabic English Parallel Text
LDC2006E24[a]	GALE Y1 - Interim Release: Translations
LDC2006E25[a]	GALE Y1 - Arabic English Parallel News Text
LDC2006E34[a]	GALE Y1 Q2 Release - Translations V2.0
LDC2006E85[a]	GALE Y1 Q3 Release - Translations
LDC2006E86[a]	GALE Y1 Q3 Release - Word Alignment
LDC2006E92[a]	GALE Y1 Q4 Release - Translations
LDC2006E93[a]	GALE Y1 Q4 Release - Word Alignment
LDC2007E07	ISI Arabic–English Automatically Extracted Parallel Text

The permissible parallel texts from the NIST MT 2008 evaluation (http://www.itl.nist.gov/iad/mig/tests/mt/2008/doc/mt08_constrained.html)
[a] Used for hierarchical phrase extraction

automatically extracted rules (rule penalty), (c) identity rules (translating a word into itself), (d) two classes of number/name translation rules, and (e) glue rules. The probability features were base-100 log-probabilities.[6] The rules were extracted from all the allowable parallel text from the NIST MT 2008 evaluation (152+175 million words of Arabic+English, in 6,561,091 parallel sentences), aligned by IBM Model 4 using GIZA++ (union of both directions). Hierarchical rules were extracted from the most in-domain corpora (4.2+5.4 million words in 170,863 parallel sentences), and phrases were extracted from the remainder. See Table 5.

Two language models were trained, with the only difference being that one was trained on data similar to the English side of the parallel text, and the other on 2 billion words of English, mainly from the LDC English Gigaword 2. Both were 5-gram models with modified Kneser-Ney smoothing, lossily compressed using a perfect-hashing scheme similar to that of Guthrie et al. (2010).

The documents of the NIST MT 2004 (newswire) and 2005 Arabic–English evaluation data were randomly partitioned into a tuning set (1178 sentences) and a development set (1298 sentences). The test data was the NIST MT 2006 Arabic–English evaluation data (NIST part, newswire and newsgroups, 1529 sentences). See Table 6.

Here, both MERT and MIRA were run on the tuning set using 20 parallel processors. MERT was stopped when the score on the tuning set stopped increasing, as is

[6] MIRA can, like the perceptron, be thought of as a gradient-descent optimization of the SVM primal objective function (generalized hinge loss). As such, it is sensitive to the relative scales of individual features. Language models tend to have large feature values, and therefore there would be large differences in feature
Footnote 6 continued
values between hypothesis translations and (what is treated as) the correct translation, which in turn would likely cause large updates to their weights; but the model is highly sensitive to changes in the language model weights. So we scale the weights of some features down by using base-100 log-probabilities.

Table 6 Training, tuning, development and test set sizes for Arabic–English translation (MIRA)

Use	Set	Size (sentences)
Training	See Table 5	6,561,091
Tuning	NIST MT04 (newswire)	1,178
Development	NIST MT05	1,298
Test	NIST MT06 (NIST part, newswire and newsgroups)	1,529

common practice; MIRA was stopped when the score on the development set stopped increasing, and after no more than 20 iterations.[7] In these runs, MERT took an average of 9 passes through the tuning set and MIRA took an average of 8 passes. For comparison, Watanabe et al. (2007) report decoding their tuning data of 663 sentences 80 times.

To obtain syntactic parses for this data, it was tokenized according to the Arabic Treebank standard using AMIRA (Diab et al. 2004), and parsed with the Stanford parser. Then, the parsing trees were forced back into the MT system's tokenization.[8]

The syntactic features in this section were organized into coarse-grained and fine-grained sets, with minor differences in implementation from the features that were used in the experiments described in Sects. 3 and 4.1. The coarse-grained feature set included, in addition to the twelve features in the baseline model, $\text{XP}^{=}$ and XP^{+}. For the fine-grained feature set, the following nonterminal labels that appear more than 100 times in the tuning data were selected: all those in XP, plus WHNP, PRT, and PRN. The labels that were excluded were mostly parts of speech, and non-constituent labels like FRAG. For each of these labels X, we added features $X^{=}$ and X^{+}.

Table 7 shows the results of the experiments with the training methods and features described above. All significance testing was performed against the first line (MERT baseline). MIRA is shown to be superior or at least competitive with MERT when both use the baseline feature set. Indeed, the MIRA system scores significantly higher on the MT06 test set; but when the test set is broken down by genre, one can see that the MIRA system does slightly worse on newswire and better on newsgroups. This is largely attributable to the fact that the MIRA translations tend to be longer than the MERT translations, perhaps due to the "unclipped" BLEU used in searching through the forest for high-BLEU or low-BLEU translations (Dreyer et al. 2007). Since the newsgroup references are longer than the newswire references, longer translations are better on newsgroups. Table 8 shows source:target length ratios.

When more features are added to the model, the two training methods diverge more sharply. When training with MERT, the coarse-grained pair of syntax features yields a small gain on MT06, but the fine-grained syntax features do not yield any further gain. Breaking down by genre reveals no gains on newswire, but increased gains with the

[7] This MIRA training policy was chosen to avoid overfitting. However, it was possible to use the tuning set for this purpose, just as with MERT: in none of these runs would this change have made more than a 0.2 BLEU difference on the development set.

[8] The only notable consequence is that proclitic Arabic prepositions were fused onto the first word of their NP object, so that the PP and NP brackets were co-extensive.

Table 7 Comparison of MERT and MIRA on various feature sets

Train	Features	#	Dev	MT06 (NIST part)			MT08[a]
			nw	nw	ng	nw+ng	
MERT	Baseline	12	52.0	50.5	32.4	44.6	40.8
	Syntax (coarse)	14	52.2	50.9	33.0^{+}	45.0^{+}	41.2
	Syntax (fine)	34	52.1	50.4	33.5^{++}	44.8	41.2
MIRA	Baseline	12	52.0	49.8^{-}	34.2^{++}	45.3^{++}	42.8
	Syntax (coarse)	14	53.3^{++}	51.1^{++}	34.6^{++}	46.3^{++}	43.3
	Syntax (fine)	34	53.1^{++}	51.3^{+}	34.5^{++}	46.4^{++}	43.3

Key: # number of features, *nw* newswire, *ng* newsgroups, $^{+}$ or $^{++}$ significantly better than MERT baseline ($p < 0.05$ or $p < 0.01$, respectively), $^{-}$ significantly worse than MERT baseline ($p < 0.05$)
[a] All models were re-created for testing on MT08, because it was added at a later stage in order to provide better comparison with Section 4.2; the new MERT-tuned models' performance on MT06 was unfortunately 0.2–0.5 BLEU points lower than the original ones

Table 8 Source:target length ratios

Train	Features	Dev	MT06 (NIST part)			MT08[a]
		nw	nw	ng	nw+ng	
MERT	Baseline	0.973	0.988	0.783	0.917	0.882
	Syntax (coarse)	0.974	0.991	0.801	0.925	0.890
	Syntax (fine)	0.978	0.994	0.815	0.932	0.920
MIRA	Baseline	1.000	1.020	0.890	0.975	0.951
	Syntax (coarse)	1.000	1.020	0.889	0.973	0.950
	Syntax (fine)	1.000	1.020	0.888	0.974	0.953

[a] All models were re-created for testing on MT08; See Table 7

fine-grained syntax features. By contrast, when the fine-grained features are trained using MIRA, they consistently yield substantial gains: over 2 BLEU points on newsgroups, and almost one point on newswire, relative to the MERT baseline (0.3 points on newsgroups, and 1.5 points on newswire, relative to the MIRA baseline). Although not clearly outperforming the coarse features, these fine-grained syntactic features combine well with other features, yielding even further gains (Chiang et al. 2008). Testing on MT08 revalidates the contribution of our approach against this stronger baseline, with either MERT or MIRA.

5 Discussion

The results in Section 4 demonstrate, to our knowledge for the first time, that significant and sometimes substantial gains over baseline can be obtained by incorporating source-side soft syntactic constraints into a state-of-the-art SCFG SMT system.

One can also see considerable consistency across multiple test sets, in terms of which constraints tend to help most.[9] In the Arabic–English task with MERT, the top eight feature combinations show some minor rank permutations between MT06 and MT08, although bigger permutations compared to MT03 (PP^{+}.$ADVP^{=}$ and all-labels$_2$ showing the largest discrepancies); and the top five single features on MT06 maintain the same ranking on MT08, with only minor permutations on MT03.

These results also provide some insight into why the original approach may have failed to yield a positive outcome. For Chinese-English (reported in Marton and Resnik (2008)), we found that when we defined finer-grained versions of the exact-match features, there was value in biasing the model to favor matching some source language constituency types. Moreover, we found that there was significant value in allowing the model to be sensitive to violations (crossing boundaries) of source language subtrees, as opposed to only exact-matching of these syntactic constituent boundaries. These results confirm that the failure of Chiang's original experiment could not be attributed solely to poor parsing quality, since in the experiments here the parser was held constant. Finer-grained features yielded higher gains in Arabic–English tasks as well, with weights tuned with either MERT or MIRA.

Looking at feature combinations, some models with non-conflated weights of both exact-match and violation-sensitive features of the same parsing label (e.g., VP_2, S_2) achieved large gains, although note that more is not necessarily better: many combinations of more features did not yield better scores, or yielded no gain at all. No conflated feature reached significance, but it is not the case that all conflated features are worse than their non-conflated same-label counterparts. For example, S_2 and S_ achieve similar scores on the Arabic MT03, and MT06 test sets—but S_ is about half a BLEU point higher than S_2 on MT08.

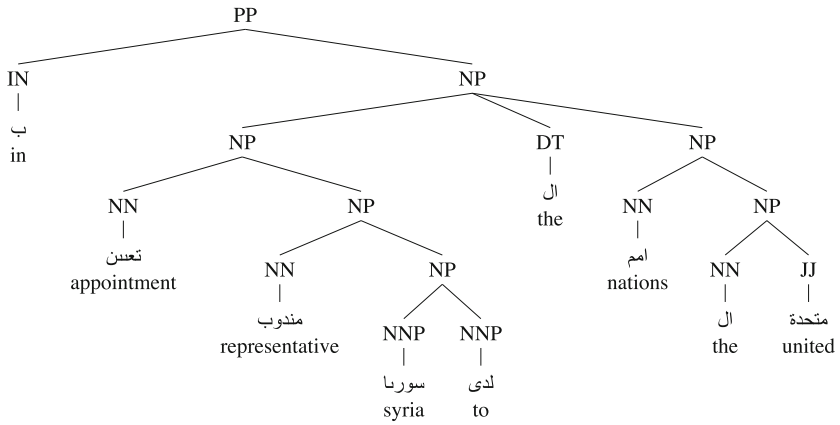
We found no simple correlation between scores of finer-grained single feature (and/or boundary type) conditions and scores of feature combination or conflation conditions. Since some combinations seem to cancel individual contributions when optimized with MERT, we can conclude that the higher the number of participant features (of the kinds described here, optimized with MERT), the more likely a cancellation effect is.

A translation example is shown in Fig. 2, where the noun phrase (NP) for *the Syrian representative* is broken in the baseline translation, but is correctly cohesively translated in the PP^{+} model. Interestingly, this model is only sensitive to PPs, and yet the soft syntactic constraints seemed to have contributed to the SMT output quality nevertheless—perhaps due to a PP that contained the NP for *the Syrian representative*, excluding the intervening NP for *the United Nations*. A more in-depth future analysis is required to better understand this effect.

Some S and VP variants seemed to do generally well. This makes sense, since clauses and verb phrases seem to correspond often on the source and target side.

We found it surprising that no NP variant yielded much gain in Arabic; it might be due to poor NP parsing quality, especially subject NPs, since the prepositional (PP) attachment problem in Arabic is very pervasive, presumably due to a higher structural

[9] Certain consistency is also observed across language pairs—see the Chinese-English experiments reported in Marton and Resnik (2008).



Reference: . . .the appointment of the syrian representative to the united nations. . .
Baseline: . . .to appoint syria to the united nations representative. . .
PP^+ . . .to appoint a representative of syria to the united nations. . .

Fig. 2 Arabic–English translation example (MERT) for the PP^+ model. The NP for *the Syrian representative* (underlined) is broken in the baseline translation, but is correctly cohesively translated in the PP^+ model, even though this model is only sensitive to PPs, and the parse is noisy. The Arabic source tree is presented word by word from left to right

ambiguity (more potential attachment sites on average) (Green et al. 2009; Carpuat et al. 2010).

Qualitatively, the tuned weights' signs $(+/-)$ indicate that the models learn to prefer obeying matching constraints, and avoiding crossing syntactic constituency boundaries. It is also worth noting that this source side soft syntactic constraints approach repeatedly yielded gains in at least three independent implementations: Marton and Resnik (2008); Chiang et al. (2008); Xiong et al. (2009)—using SCFG/MERT, SCFG/MIRA, and BTG/MERT (with inner sub-features set with MaxEnt), respectively.

The source side soft syntactic constraints approach presented here is particularly appealing because it can be used unobtrusively with any hierarchically-structured translation model. In principle, it can also be used in flat phrase-based SMT systems as well, with some modifications, as in the syntactic cohesion constraints applied by Cherry (2008) and others. It is also appealing in requiring one to parse only the development and test sets, which are relatively short, and not the training set, which would result in a considerably longer training time. The main drawback of the approach as presented in Section 4.1 was the problem of feature selection, which was removed using MIRA as presented in Section 4.2. A concern that our approach might only be effective with small training sets was largely put to rest using a large training set, also in Section 4.2.

6 Related work

The amount of work involving syntactic knowledge with SMT is vast (see the comprehensive survey by Lopez (2008), and more recently, Koehn (2010)). We will concen-

trate here on approaches that attempt to relax or soften syntactic constraints in SMT, especially those pertaining to the source language. For ease of exposition, it is useful to map the relevant literature along two axes: (1) use of syntactic parsing information of the source language versus the target language, and (2) starting from a syntactic commitment and relaxing it versus starting from a data-driven approach and adding syntactic constraints. The quadrant *adding source-side soft syntactic constraints* had been relatively unexplored before this work (Marton and Resnik 2008).

Much prior work has been on relaxation of target-side syntactic constraints in order to better exploit shallow correspondences in parallel training data. Strategies include restructuring (binarization) of target-side trees (Wang et al. 2007; DeNeefe et al. 2007), adding intermediate nodes on the fly (Marcu et al. 2006), using extended syntactic categories like NP/NN, inspired by categorial grammar (Zollmann and Venugopal 2006), or, more recently, adopting tree-insertion grammar as the underlying formalism (DeNeefe and Knight 2009).

As for relaxing syntactic constraints on the source side, Quirk et al. (2005) and Quirk and Menezes (2006) use phrasal SMT with example-based (EBMT) elements. They use source-side syntactic dependency treelets that are projected onto flat target-side phrases via unsupervised word alignments. They relax the sub-tree ordering by using an ordering model on freely ordered sub-treelets. Attempting to relax syntactic constraints on both sides, Riezler and Maxwell (2006) use LFG dependency trees on both source and target sides, and relax syntactic constraints by adding a "fragment grammar" for unparsable chunks.

Several related papers appeared concurrently with the original presentation of this work (Marton and Resnik 2008). Mi et al. (2008) relax source-side constraints by moving from a 1-best tree to a packed forest during decoding. Cherry (2008), later extended by Bach et al. (2009), incorporate source-side syntactic dependency trees as soft syntactic constraints in a weighted *syntactic cohesion* feature that penalizes translations that reorders phrases in a way inconsistent with the dependency tree.

Subsequent to our original work, Xiong et al. (2009) re-implement our XP^+ feature (see Section 3) in a phrase-reordering model using bracketing transduction grammar system (Wu 1997), and obtain over 1 BLEU point gain over their syntax-unaware baseline, in a Chinese-English translation task. They extend our features to a much more sophisticated probability model to obtain gains of up to 1.7 BLEU.

Venugopal et al. (2009) use soft syntactic constraints to make syntactic similarities between different derivations reinforce the similar parts, rather than have the entire derivations compete, as is standardly done, including the work described here. This technique alleviates the "spurious ambiguity" problem, but does not allow any new derivations.

Zhang et al. (2008) use parses of both source and target languages, and relax the syntactic constraint by allowing rules to translate *tree sequences* instead of single trees. Hanneman and Lavie (2009) relax a tree-to-tree translation system by adding a parsing tag for any non-syntactic constituent "phrase." They use it to incorporate non-syntactic phrase translations to increase coverage. Hassan et al. (2009) syntactically extend a different model—the Direct Translation Model 2, which is a linear-time decoder. They use an eager dependency parser, which linearly resolves the attachment ambigu-

ity of the next word based on combinatory categorial grammar (CCG) part-of-speech supertags.

The MIRA method we use for training and tuning our soft syntactic constraints has been further applied to training many other kinds of features (Chiang et al. 2009; Chiang 2010; Chiang et al. 2011). This line of work could be seen as a validation of the learning method. Conversely, the learning method could be seen as a vehicle for exploring new sources of information for SMT that are interesting in their own right. Our focus here is on the use of source-side syntactic parses; a comparison with features based on other sources of information is beyond the scope of this article.

7 Conclusion

When hierarchical phrase-based translation was introduced by Chiang (2005), it represented a new way to incorporate syntax into statistical MT, allowing the model to handle non-local dependencies and lexically sensitive reordering without requiring linguistically motivated parsing of either the source or target language. An approach to incorporating parser-based constituents in the model was explored briefly, treating syntactic constituency as a soft constraint, with negative results.

In the work presented here, we returned to the idea of linguistically motivated soft constraints, and we demonstrated that they can, in fact, lead to substantial improvements in translation performance when integrated into the Hiero framework. We accomplished this using constraints that not only distinguish among constituent types, but also distinguish between matching and crossing boundaries of syntactic constituents. We demonstrated gains for Chinese-English translation in Marton and Resnik (2008), and following that, we showed here substantial gains for Arabic–English translation, as well. This approach has repeatedly yielded positive results, not only when using Hiero with MERT, but also when using Hiero with MIRA, and in subsequent research by Xiong et al. (2009) using BTG with MERT.

These results contribute to a growing body of work on combining monolingually based, linguistically motivated syntactic analysis with translation models that are closely tied to observable parallel training data. Consistent with other researchers, we find that "syntactic constituency" may be too coarse a notion by itself; rather, there is value in taking a finer-grained approach, and in allowing the model to decide how far to trust each element of the syntactic analysis as part of the system's optimization process.

Acknowledgments This work was supported in part by DARPA prime agreement HR0011-06-2-0001, and in part by DARPA contract HR0011-06-C-0022 under subcontract to BBN Technologies and HR0011-06-02-001 under subcontract to IBM. The authors would like to thank the Stanford Parser team for making their parsers available. Many thanks also to Amy Weinberg and CLIP Laboratory colleagues, particularly Chris Dyer, Adam Lopez, and Smaranda Muresan; and to John DeNero, Kevin Knight, Daniel Marcu, and Fei Sha.

References

Bach N, Vogel S, Cherry C (2009) Cohesive constraints in a beam search phrase-based decoder. In: Proceedings of the North American chapter of the Association for Computational Linguistics (NAACL-HLT), Short Papers, pp 1–4

Birch A, Osborne M, Koehn P (2007) CCG supertags in factored statistical machine translation. In: Proceedings of the ACL Workshop on Statistical Machine Translation

Brown PF, Cocke J, Pietra SD, Pietra VJD, Jelinek F, Lafferty JD, Mercer RL, Roossin PS (1990) A statistical approach to machine translation. Comput Linguist 16(2):79–85

Brown PF, Pietra SAD, Pietra VJD, Mercer RL (1993) The mathematics of statistical machine translation. Comput Linguist 19(2):263–313

Carpuat M, Marton Y, Habash N (2010) Explorations in subject-verb reordering for Arabic–English statistical machine translation. In: Proceedings of the 48th Annual Conference of the Association for Computational Linguistics (ACL)

Chen SF, Goodman J (1998) An empirical study of smoothing techniques for language modeling. Technical Report TR-10-98, Computer Science Group, Harvard University

Cherry C (2008) Cohesive phrase-based decoding for statistical machine translation. In: Proceedings of the 46th Annual Conference of the Association for Computational Linguistics: Human Language Technology (ACL-HLT), pp 72–80

Chiang D (2005) A hierarchical phrase-based model for statistical machine translation. In: Proceedings of the 43rd Annual Conference of the Association for Computational Linguistics (ACL), pp 263–270

Chiang D (2007) Hierarchical phrase-based translation. Comput Linguist 33(2):201–228

Chiang D (2010) Learning to translate with source and target syntax. In: Proceedings of the 48th Annual Conference of the Association for Computational Linguistics (ACL), pp 1443–1452

Chiang D, Lopez A, Madnani N, Monz C, Resnik P, Subotin M (2005) The Hiero machine translation system: extensions, evaluation, and analysis. In: Proceedings of the Conference on Empirical Methods in Natural Language (EMNLP)–Human Language Technology (HLT), pp 779–786

Chiang D, Marton Y, Resnik P (2008) Online large-margin training of syntactic and structural translation features. In: Procedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)

Chiang D, Knight K, Wang W (2009) 11,001 new features for statistical machine translation. In: Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT), pp 218–226

Chiang D, DeNeefe S, Pust M (2011) Two easy improvements to lexical weighting. In: Proceedings of the 49th Annual Conference of the Association for Computational Linguistics (ACL), poster session

Cowan B, Kucerova I, Collins M (2006) A discriminative model for tree-to-tree translation. In: Proceedings of the Conference on Empirical Methods in Natural Language (EMNLP)

Crammer K, Singer Y (2003) Ultraconservative online algorithms for multiclass problems. J Mach Learn Res 3:951–991

DeNeefe S, Knight K (2009) Synchronous tree adjoining machine translation. In: Proceedings of the 2009 Annual Meeting of the Association for Computational Linguistics (ACL)

DeNeefe S, Knight K, Wang W, Marcu D (2007) What can syntax-based MT learn from phrase-based MT? In: Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP/CoNLL)

Diab M, Hacioglu K, Jurafsky D (2004) Automatic tagging of Arabic text: From raw text to base phrase chunks. In: Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT), pp 149–152, companion volume

Dreyer M, Hall K, Khudanpur S (2007) Comparing reordering constraints for SMT using efficient BLEU oracle computation. In: Proc. 2007 Workshop on Syntax and Structure in Statistical Translation

Eisner J (2003) Learning non-isomorphic tree mappings for machine translation. In: Proceedings of the the Annual Meeting of the Association for Computational Linguistics (ACL) Companion Volume

Fox H (2002) Phrasal cohesion and statistical machine translation. In: Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP)

Galley M, Graehl J, Knight K, Marcu D, DeNeefe S, Wang W, Thayer I (2006) Scalable inference and training of context-rich syntactic translation models. In: Proceedings of the Joint Conference of the International Committee on Computational Linguistics and the Association for Computational Linguistics

Green S, Sathi C, Manning CD (2009) NP subject detection in verb-initial Arabic clauses. In: Proceedings of the Third Workshop on Computational Approaches to Arabic Script-based Languages (CAASL3), Machine Translation Summit XII

Guthrie D, Hepple M, Liu W (2010) Efficient minimal perfect hash language models. In: Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC), Valletta, Malta, pp 2889–2896

Hanneman G, Lavie A (2009) Decoding with syntactic and non-syntactic phrases in a syntax-based machine translation system. In: Proceedings of the Third Workshop on Syntax and Structure in Statistical Translation

Hassan H, Sima'an K, Way A (2007) Integrating supertags into phrase-based statistical machine translation. In: Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL), pp 288–295

Hassan H, Sima'an K, Way A (2009) A syntactified direct translation model with linear-time decoding. In: Proceedings of the Conference on Empirical Methods in Natural Language (EMNLP), vol 3, pp 1182–1191

Klein D, Manning CD (2003a) Accurate unlexicalized parsing. In: Proceedings of the 41st Meeting of the Association for Computational Linguistics (ACL), pp 423–430

Klein D, Manning CD (2003b) Fast exact inference with a factored model for natural language parsing. Adv Neural Inf Process Syst (NIPS) 15:3–10

Koehn P (2003) Noun phrase translation. PhD thesis, University of Southern California

Koehn P (2004) Statistical significance tests for machine translation evaluation. In: Proceedings of the Conference on Empirical Methods in Natural Language (EMNLP)

Koehn P (2010) Statistical machine translation. Cambridge University Press, Cambridge

Koehn P, Hoang H (2007) Factored translation models. In: Proceedings of Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP/CoNLL), pp 868–876

Koehn P, Och FJ, Marcu D (2003) Statistical phrase-based translation. In: Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT), pp 127–133

Koehn P, Hoang H, Birch A, Callison-Burch C, Federico M, Bertoldi N, Cowan B, Shen W, Zens CMR, Dyer C, Bojar O, Constantin A, Herbst E (2007) Moses: Open source toolkit for statistical machine translation. In: Annual Meeting of the Association for Computational Linguistics (ACL), demonstration session

Lopez A (2008) Statistical machine translation. ACM Comput Surv 40(3):1–49

Marcu D, Wang W, Echihabi A, Knight K (2006) SPMT: Statistical machine translation with syntactified target language phrases. In: Proceedings of the Conference on Empirical Methods in Natural Language (EMNLP), pp 44–52

Marton Y (2009) Fine-grained linguistic soft constraints on statistical natural language processing models. Doctoral dissertation, University of Maryland, College Park

Marton Y, Resnik P (2008) Soft syntactic constraints for hierarchical phrase-based translation. In: Proceedings of the 2008 Annual Meeting of the Association for Computational Linguistics (ACL-HLT), pp 1003–1011

Mi H, Huang L, Liu Q (2008) Forest-based translation. In: Proceedings of the 2008 Annual Meeting of the Association for Computational Linguistics (ACL-HLT), pp 192–199

Och FJ (2003) Minimum error rate training in statistical machine translation. In: Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL), pp 160–167

Och FJ, Ney H (2000) Improved statistical alignment models. In: Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL), pp 440–447

Papineni K, Roukos S, Ward T, Henderson J, Reeder F (2002) Corpus-based comprehensive and diagnostic MT evaluation: Initial Arabic, Chinese, French, and Spanish results. In: Proceedings of the 2002 Annual Meeting of the Association for Computational Linguistics (ACL-HLT), pp 124–127

Quirk C, Menezes A (2006) Dependency treelet translation: the convergence of statistical and example-based machine translation?. Mach Transl 20:43–65

Quirk C, Menezes A, Cherry C (2005) Dependency treelet translation: Syntactically informed phrasal SMT. In: Proceedings of the 2005 Annual Meeting of the Association for Computational Linguistics (ACL)

Riezler S, Maxwell J (2006) Grammatical machine translation. In: Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)

Stolcke A (2002) SRILM—an extensible language modeling toolkit. In: Proceedings of the International Conference on Spoken Language Processing, vol 2, pp 901–904

Venugopal A, Zollmann A, Smith N, Vogel S (2009) Preference grammars: Softening syntactic constraints to improve statistical machine translation. In: Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL)

Wang W, Knight K, Marcu D (2007) Binarizing syntax trees to improve syntax-based machine translation accuracy. In: Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP/CoNLL)

Watanabe T, Suzuki J, Tsukuda H, Isozaki H (2007) Online large-margin training for statistical machine translation. In: Proceedings of the Conference on Empirical Methods in Natural Language (EMNLP)

Wu D (1997) Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. Comput Linguist 23:377–404

Xiong D, Zhang M, Aw A, Li H (2009) A syntax-driven bracketing model for phrase-based translation. In: Proceedings of the 47th Annual Conference of the Association for Computational Linguistics (ACL)

Zhang M, Jiang H, Aw A, Li H, Tan CL, Li S (2008) A tree sequence alignment-based tree-to-tree translation model. In: Proceedings of the 2008 Annual Meeting of the Association for Computational Linguistics (ACL-HLT), pp 559–567

Zollmann A, Venugopal A (2006) Syntax augmented machine translation via chart parsing. In: Proceedings of the SMT Workshop at the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)