Predicting Target Language CCG Supertags Improves Neural Machine Translation

Maria Nădejde[1] and **Siva Reddy**[1] and **Rico Sennrich**[1] and **Tomasz Dwojak**[1,2]
Marcin Junczys-Dowmunt[2] and **Philipp Koehn**[3] and **Alexandra Birch**[1]
[1]School of Informatics, University of Edinburgh
[2]Adam Mickiewicz University
[3]Dep. of Computer Science, Johns Hopkins University
{m.nadejde,siva.reddy, rico.sennrich, a.birch}@ed.ac.uk
{t.dwojak,junczys}@amu.edu.pl,phi@jhu.edu

Abstract

Neural machine translation (NMT) models are able to partially learn syntactic information from sequential lexical information. Still, some complex syntactic phenomena such as prepositional phrase attachment are poorly modeled. This work aims to answer two questions: 1) Does explicitly modeling target language syntax help NMT? 2) Is tight integration of words and syntax better than multitask training? We introduce syntactic information in the form of CCG supertags in the decoder, by interleaving the target supertags with the word sequence. Our results on WMT data show that explicitly modeling target-syntax improves machine translation quality for German→English, a high-resource pair, and for Romanian→English, a low-resource pair and also several syntactic phenomena including prepositional phrase attachment. Furthermore, a tight coupling of words and syntax improves translation quality more than multitask training. By combining target-syntax with adding source-side dependency labels in the embedding layer, we obtain a total improvement of 0.9 BLEU for German→English and 1.2 BLEU for Romanian→English.

1 Introduction

Sequence-to-sequence neural machine translation (NMT) models (Sutskever et al., 2014; Cho et al., 2014b; Bahdanau et al., 2015) are state-of-the-art on a multitude of language-pairs (Sennrich et al., 2016a; Junczys-Dowmunt et al., 2016). Part of the appeal of neural models is that they can learn to implicitly model phenomena which underlie high quality output, and some syntax is indeed cap-

tured by these models. In a detailed analysis, Bentivogli et al. (2016) show that NMT significantly improves over phrase-based SMT, in particular with respect to morphology and word order, but that results can still be improved for longer sentences and complex syntactic phenomena such as prepositional phrase (PP) attachment. Another study by Shi et al. (2016) shows that the encoder layer of NMT partially learns syntactic information about the source language, however complex syntactic phenomena such as coordination or PP attachment are poorly modeled.

Recent work which incorporates additional source-side linguistic information in NMT models (Luong et al., 2016; Sennrich and Haddow, 2016) show that even though neural models have strong learning capabilities, explicit features can still improve translation quality. In this work, we examine the benefit of incorporating global syntactic information on the target-side. We also address the question of how best to incorporate this information. For language pairs where syntactic resources are available on both the source and target-side, we show that approaches to incorporate source syntax and target syntax are complementary.

We propose a method for tightly coupling words and syntax by interleaving the target syntactic representation with the word sequence. We compare this to loosely coupling words and syntax using a multitask solution, where the shared parts of the model are trained to produce either a target sequence of words or supertags in a similar fashion to Luong et al. (2016).

We use CCG syntactic categories (Steedman, 2000), also known as *supertags*, to represent syntax explicitly. Supertags provide global syntactic information locally at the lexical level. They encode subcategorization information, capturing short and long range dependencies and attach-

ments, and also tense and morphological aspects of the word in a given context. Consider the sentence in Figure 1. This sentence contains two PP attachments and could lead to several disambiguation possibilities (*"in"* can attach to *"Netanyahu"* or *"receives"*, and *"of"* can attach to *"capital"*, *"Netanyahu"* or *"receives"*). These alternatives may lead to different translations in other languages. However the supertag $((S[dcl]\backslash NP)/PP)/NP$ of *"receives"* indicates that the preposition *"in"* attaches to the verb, and the supertag $(NP\backslash NP)/NP$ of *"of"* indicates that it attaches to *"capital"*, thereby resolving the ambiguity.

Our research contributions are as follows:

- We propose a novel approach to integrating target syntax at word level in the decoder, by interleaving CCG supertags in the target word sequence.

- We show that the target language syntax improves translation quality for German→English and Romanian→English as measured by BLEU. Our results suggest that a tight coupling of target words and syntax (by interleaving) improves translation quality more than the decoupled signal from multitask training.

- We show that incorporating source-side linguistic information is complimentary to our method, further improving the translation quality.

- We present a fine-grained analysis of SNMT and show consistent gains for different linguistic phenomena and sentence lengths.

2 Related work

Syntax has helped in statistical machine translation (SMT) to capture dependencies between distant words that impact morphological agreement, subcategorisation and word order (Galley et al., 2004; Menezes and Quirk, 2007; Williams and Koehn, 2012; Nadejde et al., 2013; Sennrich, 2015; Nadejde et al., 2016a,b; Chiang, 2007). There has been some work in NMT on modeling source-side syntax implicitly or explicitly. Kalchbrenner and Blunsom (2013); Cho et al. (2014a) capture the hierarchical aspects of language implicitly by using convolutional neural networks, while Eriguchi et al. (2016) use the parse tree of the source sentence to guide the recurrence and attention model in tree-to-sequence NMT. Luong

et al. (2016) co-train a translation model and a source-side syntactic parser which share the encoder. Our multitask models extend their work to attention-based NMT models and to predicting target-side syntax as the secondary task. Sennrich and Haddow (2016) generalize the embedding layer of NMT to include explicit linguistic features such as dependency relations and part-of-speech tags and we use their framework to show source and target syntax provide complementary information.

Applying more tightly coupled linguistic factors on the target for NMT has been previously investigated. Niehues et al. (2016) proposed a factored RNN-based language model for re-scoring an n-best list produced by a phrase-based MT system. In recent work, Martínez et al. (2016) implemented a factored NMT decoder which generated both lemmas and morphological tags. The two factors were then post-processed to generate the word form. Unfortunately no real gain was reported for these experiments. Concurrently with our work, Aharoni and Goldberg (2017) proposed serializing the target constituency trees and Eriguchi et al. (2017) model target dependency relations by augmenting the NMT decoder with a RNN grammar (Dyer et al., 2016). In our work, we use CCG supertags which are a more compact representation of global syntax. Furthermore, we do not focus on model architectures, and instead we explore the more general problem of including target syntax in NMT: comparing tightly and loosely coupled syntactic information and showing source and target syntax are complementary.

Previous work on integrating CCG supertags in factored phrase-based models (Birch et al., 2007) made strong independence assumptions between the target word sequence and the CCG categories. In this work we take advantage of the expressive power of recurrent neural networks to learn representations that generate both words and CCG supertags, conditioned on the entire lexical and syntactic target history.

3 Modeling Syntax in NMT

CCG is a lexicalised formalism in which words are assigned with syntactic categories, i.e., supertags, that indicate context-sensitive morpho-syntactic properties of a word in a sentence. The combinators of CCG allow the supertags to capture global syntactic constraints locally. Though NMT

Source-side									
BPE:	Obama	receives	Net+	an+	yahu	in	the	capital	of	USA
IOB:	O	O	B	I	E	O	O	O	O	O
CCG:	NP	((S[dcl]\NP)/PP)/NP	NP	NP	NP	PP/NP	NP/N	N	(NP\NP)/NP	NP
Target-side									

NP Obama ((S[dcl]\NP)/PP)/NP receives NP Net+ an+ yahu PP/NP in NP/N the N capital (NP\NP)/NP of NP USA

Figure 1: Source and target representation of syntactic information in syntax-aware NMT.

captures long range dependencies using long-term memory, short-term memory is cheap and reliable. Supertags can help by allowing the model to rely more on local information (short-term) and not having to rely heavily on long-term memory.

Consider a decoder that has to generate the following sentences:

1. What$_{(S[wq]/(S[q]/NP))/N}$ city is$_{(S[q]/PP)/NP}$ the Taj Mahal in?

2. Where$_{S[wq]/(S[q]/NP)}$ is$_{(S[q]/NP)/NP}$ the Taj Mahal?

If the decoding starts with predicting "*What*", it is ungrammatical to omit the preposition "*in*", and if the decoding starts with predicting "*Where*", it is ungrammatical to predict the preposition. Here the decision to predict "*in*" depends on the first word, a long range dependency. However if we rely on CCG supertags, the supertags of both these sequences look very different. The supertag (S[q]/PP)/NP for the verb "*is*" in the first sentence indicates that a preposition is expected in future context. Furthermore it is likely to see this particular supertag of the verb in the context of (S[wq]/(S[q]/NP))/N but it is unlikely in the context of S[wq]/(S[q]/NP). Therefore a succession of local decisions based on CCG supertags will result in the correct prediction of the preposition in the first sentence, and omitting the preposition in the second sentence. Since the vocabulary of CCG supertags is much smaller than that of possible words, the NMT model will do a better job at generalizing over and predicting the correct CCG supertags sequence.

CCG supertags also help during encoding if they are given in the input, as we saw with the case of PP attachment in Figure 1. Translation of the correct verb form and agreement can be improved with CCG since supertags also encode tense, morphology and agreements. For example, in the sentence "*It is going to rain*", the supertag (S[ng]\NP[expl])/(S[to]\NP) of "*going*"

indicates the current word is a verb in continuous form looking for an infinitive construction on the right, and an expletive pronoun on the left.

We explore the effect of target-side syntax by using CCG supertags in the decoder and by combining these with source-side syntax in the encoder, as follows.

Baseline decoder The baseline decoder architecture is a conditional GRU with attention ($cGRU_{attn}$) as implemented in the Nematus toolkit (Sennrich et al., 2017). The decoder is a recursive function computing a hidden state s_j at each time step $j \in [1, T]$ of the target recurrence. This function takes as input the previous hidden state s_{j-1}, the embedding of the previous target word y_{j-1} and the output of the attention model c_j. The attention model computes a weighted sum over the hidden states $h_i = [\overrightarrow{h_i}; \overleftarrow{h_i}]$ of the bi-directional RNN encoder. The function g computes the intermediate representation t_j and passes this to a *softmax* layer which first applies a linear transformation (W_o) and then computes the probability distribution over the target vocabulary. The training objective for the entire architecture is minimizing the discrete cross-entropy, therefore the loss l is the negative log-probability of the reference sentence.

$$s'_j = GRU_1(y_{j-1}, s_{j-1}) \tag{1}$$

$$c_j = ATT([h_1; ...; h_{|x|}], s'_j) \tag{2}$$

$$s_j = cGRU_{attn}(y_{j-1}, s_{j-1}, c_j) \tag{3}$$

$$t_j = g(y_{j-1}, s_j, c_j) \tag{4}$$

$$p_y = \prod_{j=1}^{T} p(y_j|x, y_{1:j-1}) = \prod_{j=1}^{T} softmax(t_j W_o) \tag{5}$$

$$l = -log(p_y) \tag{6}$$

Target-side syntax When modeling the target-side syntactic information we consider different
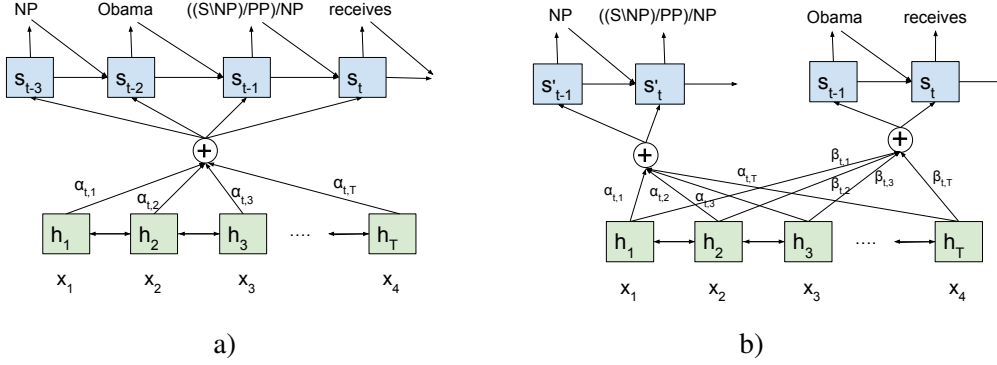
70

Figure 2: Integrating target syntax in the NMT decoder: a) interleaving and b) multitasking.

strategies of coupling the CCG supertags with the translated words in the decoder: interleaving and multitasking with shared encoder. In Figure 2 we represent graphically the differences between the two strategies and in the next paragraphs we formalize them.

- **Interleaving** In this paper we propose a tight integration in the decoder of the syntactic representation and the surface forms. Before each word of the target sequence we include its supertag as an extra token. The new target sequence y' will have the length $2T$, where T is the number of target words. With this representation, a single decoder learns to predict both the target supertags and the target words conditioned on previous syntactic and lexical context. We do not make changes to the baseline NMT decoder architecture, keeping equations (1) - (6) and the corresponding set of parameters unchanged. Instead, we augment the target vocabulary to include both words and CCG supertags. This results in a shared embedding space and the following probability of the target sequence y', where y'_j can be either a word or a tag:

$$y' = y_1^{tag}, y_1^{word}, ..., y_T^{tag}, y_T^{word} \qquad (7)$$

$$p_{y'} = \prod_{j}^{2T} p(y'_j | x, y'_{1:j-1}) \qquad (8)$$

At training time we pre-process the target sequence to add the syntactic annotation and then split only the words into *byte-pair-encoding* (BPE) (Sennrich et al., 2016b) sub-units. At

testing time we delete the predicted CCG supertags to obtain the final translation. Figure 1 gives an example of the target-side representation in the case of interleaving. The supertag *NP* corresponding to the word *Netanyahu* is included only once before the three BPE subunits *Net+ an+ yahu*.

- **Multitasking – shared encoder** A loose coupling of the syntactic representation and the surface forms can be achieved by co-training a translation model with a secondary prediction task, in our case CCG supertagging. In the multitask framework (Luong et al., 2016) the encoder part is shared while the decoder is different for each of the prediction tasks: translation and tagging. In contrast to Luong et al., we train a separate attention model for each task and perform multitask learning with target syntax. The two decoders take as input the same source context, represented by the encoder's hidden states $h_i = [\overrightarrow{h_i}; \overleftarrow{h_i}]$. However, each task has its own set of parameters associated with the five components of the decoder: GRU_1, ATT, $cGRU_{att}$, g, $softmax$. Furthermore, the two decoders may predict a different number of target symbols, resulting in target sequences of different lengths T_1 and T_2. This results in two probability distributions over separate target vocabularies for the words and the tags:

$$p_y^{word} = \prod_{j}^{T_1} p(y_j^{word} | x, y_{1:j-1}^{word}) \qquad (9)$$

$$p_y^{tag} = \prod_{k}^{T_2} p(y_k^{tag} | x, y_{1:k-1}^{tag}) \qquad (10)$$

71

The final loss is the sum of the losses for the two decoders:

$$l = -(log(p_y^{word}) + log(p_y^{tag})) \quad (11)$$

We use EasySRL to label the English side of the parallel corpus with CCG supertags[1] instead of using a corpus with gold annotations as in Luong et al. (2016).

Source-side syntax – shared embedding   While our focus is on target-side syntax, we also experiment with including source-side syntax to show that the two approaches are complementary.

Sennrich and Haddow propose a framework for including source-side syntax as extra features in the NMT encoder. They extend the model of Bahdanau et al. by learning a separate embedding for several source-side features such as the word itself or its part-of-speech. All feature embeddings are concatenated into one embedding vector which is used in all parts of the encoder model instead of the word embedding. When modeling the source-side syntactic information, we include the CCG supertags or dependency labels as extra features. The baseline features are the subword units obtained using BPE together with the annotation of the subword structure using IOB format by marking if a symbol in the text forms the beginning (B), inside (I), or end (E) of a word. A separate tag (O) is used if a symbol corresponds to the full word. The word level supertag is replicated for each BPE unit. Figure 1 gives an example of the source-side feature representation.

4   Experimental Setup and Evaluation

4.1   Data and methods

We train the neural MT systems on all the parallel data available at WMT16 (Bojar et al., 2016) for the German↔English and Romanian↔English language pairs. The English side of the training data is annotated with CCG lexical tags[2] using EasySRL (Lewis et al., 2015) and the available pre-trained model[3]. Some longer sentences cannot be processed by the parser and therefore we eliminate them from our training and test data. We report the sentence counts for the filtered data

	train	dev	test
DE-EN	4,468,314	2,986	2,994
RO-EN	605,885	1,984	1,984

Table 1:   Number of sentences in the training, development and test sets.

sets in Table 1. Dependency labels are annotated with ParZU (Sennrich et al., 2013) for German and SyntaxNet (Andor et al., 2016) for Romanian.

All the neural MT systems are attentional encoder-decoder networks (Bahdanau et al., 2015) as implemented in the Nematus toolkit (Sennrich et al., 2017).[4] We use similar hyper-parameters to those reported by (Sennrich et al., 2016a; Sennrich and Haddow, 2016) with minor modifications: we used mini-batches of size 60 and Adam optimizer (Kingma and Ba, 2014). We select the best single models according to BLEU on the development set and use the four best single models for the ensembles.

To show that we report results over strong baselines, table 2 compares the scores obtained by our baseline system to the ones reported in Sennrich et al. (2016a). We normalize diacritics[5] for the English→Romanian test set. We did not remove or normalize Romanian diacritics for the other experiments reported in this paper. Our baseline systems are generally stronger than Sennrich et al. (2016a) due to training with a different optimizer for more iterations.

	This work	Sennrich et. al
DE→EN	31.0	28.5
EN→DE	27.8	26.8
RO→EN	28.0	27.8
EN→RO[1]	25.6	23.9

Table 2:   Comparison of baseline systems in this work and in Sennrich et al. (2016a). Case-sensitive BLEU scores reported over newstest2016 with *mteval-13a.perl*. [1]Normalized diacritics.

During training we validate our models with BLEU (Papineni et al., 2002) on development sets: newstest2013 for German↔English and newsdev2016 for Romanian↔English. We evaluate the systems on newstest2016 test sets for both lan-

---

[1]We use the same data and annotations for the *interleaving* approach.

[2]The CCG tags include features such as the verb tense (e.g. [ng] for continuous form) or the sentence type (e.g. [pss] for passive).

[3]https://github.com/uwnlp/EasySRL

[4]https://github.com/rsennrich/nematus

[5]There are different encodings for letters with cedilla (ş,ţ) used interchangeably throughout the corpus. https://en.wikipedia.org/wiki/Romanian_alphabet#ISO_8859

guage pairs and use bootstrap resampling (Riezler and Maxwell, 2005) to test statistical significance. We compute BLEU with *multi-bleu.perl* over tokenized sentences both on the development sets, for early stopping, and on the test sets for evaluating our systems.

Words are segmented into sub-units that are learned jointly for source and target using BPE (Sennrich et al., 2016b), resulting in a vocabulary size of 85,000. The vocabulary size for CCG supertags was 500.

For the experiments with source-side features we use the BPE sub-units and the IOB tags as baseline features. We keep the total word embedding size fixed to 500 dimensions. We allocate 10 dimensions for dependency labels when using these as source-side features and when using source-side CCG supertags we allocate 135 dimensions.

The *interleaving* approach to integrating target syntax increases the length of the target sequence. Therefore, at training time, when adding the CCG supertags in the target sequence we increase the maximum length of sentences from 50 to 100. On average, the length of English sentences for newstest2013 in BPE representation is 22.7, while the average length when adding the CCG supertags is 44. Increasing the length of the target recurrence results in larger memory consumption and slower training.[6]. At test time, we obtain the final translation by post-processing the predicted target sequence to remove the CCG supertags.

4.2 Results

In this section, we first evaluate the syntax-aware NMT model (SNMT) with target-side CCG supertags as compared to the baseline NMT model described in the previous section (Bahdanau et al., 2015; Sennrich et al., 2016a). We show that our proposed method for tightly coupling target syntax via *interleaving*, improves translation for both German→English and Romanian→English while the *multitasking* framework does not. Next, we show that SNMT with target-side CCG supertags can be complemented with source-side dependencies, and that combining both types of syntax brings the most improvement. Finally, our experiments with source-side CCG supertags confirm that global syntax can improve translation either

---

[6]Roughly 10h30 per 100,000 sentences (20,000 batches) for SNMT compared to 6h for NMT.

as extra information in the encoder or in the decoder.

Target-side syntax We first evaluate the impact of target-side CCG supertags on overall translation quality. In Table 3 we report results for German→English, a high-resource language pair, and for Romanian→English, a low-resource language pair. We report BLEU scores for both the best single models and ensemble models. However, we will only refer to the results with ensemble models since these are generally better.

The SNMT system with target-side syntax improves BLEU scores by 0.9 for Romanian→English and by 0.6 for German→English. Although the training data for German→English is large, the CCG supertags still improve translation quality. These results suggest that the baseline NMT decoder benefits from modeling the global syntactic information locally via supertags.

Next, we evaluate whether there is a benefit to tight coupling between the target word sequence and syntax, as apposed to loose coupling. We compare our method of *interleaving* the CCG supertags with *multitasking*, which predicts target CCG supertags as a secondary task. The results in Table 3 show that the multitask approach does not improve BLEU scores for German→English, which exhibits long distance word reordering. For Romanian→English, which exhibits more local word reordering, multitasking improves BLEU by 0.6 relative to the baseline. In contrast, the *interleaving* approach improves translation quality for both language pairs and to a larger extent. Therefore, we conclude that a tight integration of the target syntax and word sequence is important. Conditioning the prediction of words on their corresponding CCG supertags is what sets SNMT apart from the multitasking approach.

Source-side and target-side syntax We now show that our method for integrating target-side syntax can be combined with the framework of Sennrich and Haddow (2016) for integrating source-side linguistic information, leading to further improvement in translation quality. We evaluate the syntax-aware NMT system, with CCG supertags as target-syntax and dependency labels as source-syntax. While the dependency labels do not encode global syntactic information, they disambiguate the grammatical function of words. Ini-

model	syntax	strategy	German→English		Romanian→English	
			single	ensemble	single	ensemble
NMT	-	-	31.0	32.1	28.1	28.4
SNMT	target – CCG	interleaving	32.0	**32.7***	29.2	**29.3****
Multitasking	target – CCG	shared encoder	31.4	32.0	28.4	29.0*
SNMT	source – dep	shared embedding	31.4	32.2	28.2	28.9
	+ target – CCG	+ interleaving	32.1	**33.0****	29.1	**29.6****

Table 3: Experiments with target-side syntax for German→English and Romanian→English. BLEU scores reported for baseline NMT, syntax-aware NMT (SNMT) and multitasking. The SNMT system is also combined with source dependencies. Statistical significance is indicated with * $p < 0.05$ and ** $p < 0.01$, when comparing against the NMT baseline.

tially, we had intended to use global syntax on the source-side as well for German→English, however the German CCG tree-bank is still under development.

From the results in Table 3 we first observe that for German→English the source-side dependency labels improve BLEU by only 0.1, while Romanian→English sees an improvement of 0.5. Source-syntax may help more for Romanian→English because the training data is smaller and the word order is more similar between the source and target languages than it is for German→English.

For both language pairs, target-syntax improves translation quality more than source-syntax. However, target-syntax is complemented by source-syntax when used together, leading to a final improvement of 0.9 BLEU points for German→English and 1.2 BLEU points for Romanian→English.

Finally, we show that CCG supertags are also an effective representation of global-syntax when used in the encoder. In Table 4 we present results for using CCG supertags as source-syntax in the embedding layer. Because we have CCG annotations only for English, we reverse the translation directions and report BLEU scores for English→German and English→Romanian. The BLEU scores reported are for the ensemble models over newstest2016.

For English→German BLEU increases by 0.7 points and for English→Romanian by 0.5 points. In contrast, Sennrich and Haddow (2016) obtain an improvement of only 0.2 for English→German using dependency labels which encode only the grammatical function of words. These results confirm that representing global syntax in the encoder provides complementary information that

model	syntax	EN→DE	EN→RO
NMT	-	28.3	25.6
SNMT	source – CCG	**29.0***	**26.1***

Table 4: Results for English→German and English→Romanian with source-side syntax. The SNMT system uses the CCG supertags of the source words in the embedding layer. *$p < 0.05$.

the baseline NMT model is not able to learn from the source word sequence alone.

4.3 Analyses by sentence type

In this section, we make a finer grained analysis of the impact of target-side syntax by looking at a breakdown of BLEU scores with respect to different linguistic constructions and sentence lengths[7].

We classify sentences into different linguistic constructions based on the CCG supertags that appear in them, e.g., the presence of category (NP\NP)/(S/NP) indicates a subordinate construction. Figure 3 a) shows the difference in BLEU points between the syntax-aware NMT system and the baseline NMT system for the following linguistic constructions: coordination *(conj)*, control and raising *(control)*, prepositional phrase attachment *(pp)*, questions and subordinate clauses *(subordinate)*. In the figure we use the symbol "*" to indicate that syntactic information is used on the target (eg. de-en*), or both on the source and target (eg. *de-en*). We report the number of sentences for each category in Table 5.

With target-syntax, we see consistent improvements across all linguistic constructions for Romanian→English and across all but *control and raising* for German→English. In particular, the in-

---
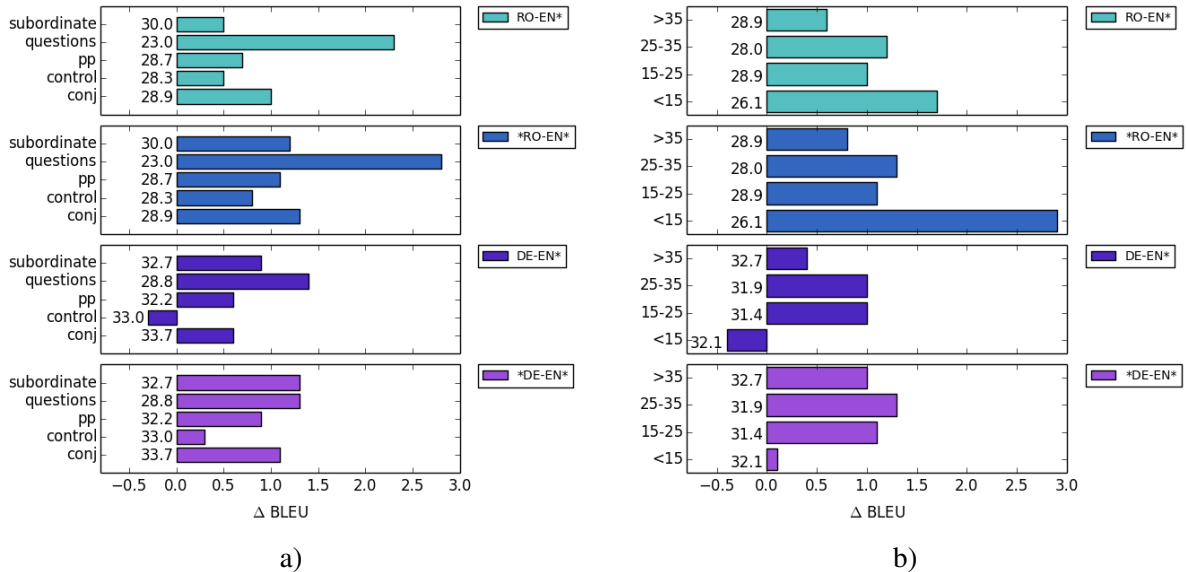[7]Document-level BLEU is computed over each subset of sentences.

Figure 3: Difference in BLEU points between SNMT and NMT, relative to baseline NMT scores, with respect to a) linguistic constructs and b) sentence lengths. The numbers attached to the bars represent the BLEU score for the baseline NMT system. The symbol * indicates that syntactic information is used on the target (eg. de-en*), or both on the source and target (eg. *de-en*)

	sub.	qu.	pp	contr.	conj
RO↔EN	742	90	1,572	415	845
DE↔EN	936	114	2,321	546	1,129

Table 5: Sentence counts for different linguistic constructions.

crease in BLEU scores for the *prepositional phrase* and *subordinate* constructions suggests that target word order is improved.

For German→English, there is a small decrease in BLEU for the *control and raising* constructions when using target-syntax alone. However, source-syntax adds complementary information to target-syntax, resulting in a small improvement for this category as well. Moreover, combining source and target-syntax increases translation quality across all linguistic constructions as compared to NMT and SNMT with target-syntax alone. For Romanian→English, combining source and target-syntax brings an additional improvement of 0.7 for *subordinate* constructs and 0.4 for *prepositional phrase attachment*. For German→English, on the same categories, there is an additional improvement of 0.4 and 0.3 respectively. Overall, BLEU scores improve by more than 1 BLEU point for most linguistic constructs and for

both language pairs.

Next, we compare the systems with respect to sentence length. Figure 3 b) shows the difference in BLEU points between the syntax-aware NMT system and the baseline NMT system with respect to the length of the source sentence measured in BPE sub-units. We report the number of sentences for each category in Table 6.

	<15	15-25	25-35	>35
RO↔EN	491	540	433	520
DE↔EN	918	934	582	560

Table 6: Sentence counts for different sentence lengths.

With target-syntax, we see consistent improvements across all sentence lengths for Romanian→English and across all but short sentences for German→English. For German→English there is a decrease in BLEU for sentences up to 15 words. Since the German→English training data is large, the baseline NMT system learns a good model for short sentences with local dependencies and without subordinate or coordinate clauses. Including extra CCG supertags increases the target sequence without adding information about complex lin-

	DE - EN Question
Source	Oder wollen Sie herausfinden , **über** was andere reden ?
Ref.	Or do you want to find out what others are talking **about** ?
NMT	Or would you like to find out **about** what others are talking **about** ?
SNMT	Or do you want to find out what$_{NP/(S[dcl]/NP)}$ others are$_{(S[dcl]\backslash NP)/(S[ng]\backslash NP)}$ talking$_{(S[ng]\backslash NP)/PP}$ **about**$_{PP/NP}$?

	DE - EN Subordinate
Source	...dass die Polizei jetzt sagt , ..., und dass Lamb in seinem Notruf **Prentiss zwar als seine Frau bezeichnete** ...
Ref.	...that police are now saying ..., and that while Lamb **referred to Prentiss as his wife** in the 911 call ...
NMT	...police are now saying ..., and that in his emergency call **Prentiss he called his wife** ...
SNMT	...police are now saying ..., and that lamb , in his emergency call , **described**$_{((S[dcl]\backslash NP)/PP)/NP}$ **Prentiss as his wife** ....

Figure 4: Comparison of baseline NMT and SNMT with target syntax for German→English.

guistic phenomena. However, when using both source and target syntax, the effect on short sentences disappears. For Romanian→English there is also a large improvement on short sentences when combining source and target syntax: 2.9 BLEU points compared to the NMT baseline and 1.2 BLEU points compared to SNMT with target-syntax alone.

With both source and target-syntax, translation quality increases across all sentence lengths as compared to NMT and SNMT with target-syntax alone. For German→English sentences that are more than 35 words, we see again the effect of increasing the target sequence by adding CCG supertags. Target-syntax helps, however BLEU improves by only 0.4, compared to 0.9 for sentences between 15 and 35 words. With both source and target syntax, BLEU improves by 0.8 for sentences with more than 35 words. For Romanian→English we see a similar result for sentences with more than 35 words: target-syntax improves BLEU by 0.6, while combining source and target syntax improves BLEU by 0.8. These results confirm as well that source-syntax adds complementary information to target-syntax and mitigates the problem of increasing the target sequence.

4.4 Discussion

Our experiments demonstrate that target-syntax improves translation for two translation directions: German→English and Romanian→English. Our proposed method predicts the target words together with their CCG supertags.

Although the focus of this paper is not improving CCG tagging, we can also measure that SNMT is accurate at predicting CCG supertags. We compare the CCG sequence predicted by the SNMT models with that predicted by EasySRL

and obtain the following accuracies: 93.2 for Romanian→English, 95.6 for German→English, 95.8 for German→English with both source and target syntax.[8]

We conclude by giving a couple of examples in Figure 4 for which the SNMT system with target syntax produced more grammatical translations than the baseline NMT system.

In the example **DE-EN Question** the baseline NMT system translates the preposition *"über"* twice as *"about"*. The SNMT system with target syntax predicts the correct CCG supertag for *"what"* which expects to be followed by a sentence and not a preposition: NP/(S[dcl]/NP). Therefore the SNMT correctly re-orders the preposition *"about"* at the end of the question.

In the example **DE-EN Subordinate** the baseline NMT system fails to correctly attach *"Prentiss"* as an object and *"his wife"* as a modifier to the verb *"called (bezeichnete)"* in the subordinate clause. In contrast the SNMT system predicts the correct sub-categorization frame of the verb *"described"* and correctly translates the entire predicate-argument structure.

5 Conclusions

This work introduces a method for modeling explicit target-syntax in a neural machine translation system, by interleaving target words with their corresponding CCG supertags. Earlier work on syntax-aware NMT mainly modeled syntax in the encoder, while our experiments suggest modeling syntax in the decoder is also useful. Our results show that a tight integration of syntax in the decoder improves translation quality for both

---

[8]The multitasking model predicts a different number of CCG supertags than the number of target words. For the sentences where these numbers match, the CCG supetagging accuracy is 73.2.

German→English and Romanian→English language pairs, more so than a loose coupling of target words and syntax as in multitask learning. Finally, by combining our method for integrating target-syntax with the framework of Sennrich and Haddow (2016) for source-syntax we obtain the most improvement over the baseline NMT system: 0.9 BLEU for German→English and 1.2 BLEU for Romanian→English. In particular, we see large improvements for longer sentences involving syntactic phenomena such as subordinate and coordinate clauses and prepositional phrase attachment. In future work, we plan to evaluate the impact of target-syntax when translating into a morphologically rich language, for example by using the Hindi CCGBank (Ambati et al., 2016).

Acknowledgements

References

Roee Aharoni and Yoav Goldberg. 2017. Towards string-to-tree neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Vancouver, Canada. Association for Computational Linguistics.

Bharat Ram Ambati, Tejaswini Deoskar, and Mark Steedman. 2016. Hindi CCGbank: CCG Treebank from the Hindi Dependency Treebank. In *Language Resources and Evaluation*.

Daniel Andor, Chris Alberti, David Weiss, Aliaksei Severyn, Alessandro Presta, Kuzman Ganchev, Slav Petrov, and Michael Collins. 2016. Globally normalized transition-based neural networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2442–2452, Berlin, Germany. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Luisa Bentivogli, Arianna Bisazza, Mauro Cettolo, and Marcello Federico. 2016. Neural versus phrase-based machine translation quality: a case study. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 257–267.

Alexandra Birch, Miles Osborne, and Philipp Koehn. 2007. Ccg supertags in factored statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 9–16, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation*, pages 131–198, Berlin, Germany. Association for Computational Linguistics.

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

Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014a. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, Doha, Qatar. Association for Computational Linguistics.

Kyunghyun Cho, Bart van Merriënboer, Çağlar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014b. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

Chris Dyer, Adhiguna Kuncoro, Miguel Ballesteros, and Noah A. Smith. 2016. Recurrent neural network grammars. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 199–209, San Diego, California. Association for Computational Linguistics.

Akiko Eriguchi, Kazuma Hashimoto, and Yoshimasa Tsuruoka. 2016. Tree-to-sequence attentional neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 823–833, Berlin, Germany. Association for Computational Linguistics.

Akiko Eriguchi, Yoshimasa Tsuruoka, and Kyunghyun Cho. 2017. Learning to parse and translate improves neural machine translation. In *Proceedings of the*

55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Vancouver, Canada. Association for Computational Linguistics.

Michel Galley, Mark Hopkins, Kevin Knight, and Daniel Marcu. 2004. What's in a translation rule? In *Proceedings of Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics*, HLT-NAACL '04.

Marcin Junczys-Dowmunt, Tomasz Dwojak, and Hieu Hoang. 2016. Is Neural Machine Translation Ready for Deployment? A Case Study on 30 Translation Directions. In *Proceedings of the IWSLT 2016*.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709, Seattle, Washington, USA. Association for Computational Linguistics.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Mike Lewis, Luheng He, and Luke Zettlemoyer. 2015. Joint a* ccg parsing and semantic role labelling. In *Empirical Methods in Natural Language Processing*.

Minh-Thang Luong, Quoc V Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2016. Multi-task sequence to sequence learning. In *Proceedings of International Conference on Learning Representations (ICLR 2016)*.

Mercedes García Martínez, Loïc Barrault, and Fethi Bougares. 2016. Factored Neural Machine Translation Architectures. In *International Workshop on Spoken Language Translation (IWSLT'16)*.

Arul Menezes and Chris Quirk. 2007. Using dependency order templates to improve generality in translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 1–8.

Maria Nadejde, Alexandra Birch, and Philipp Koehn. 2016a. Modeling selectional preferences of verbs and nouns in string-to-tree machine translation. In *Proceedings of the First Conference on Machine Translation*, pages 32–42, Berlin, Germany. Association for Computational Linguistics.

Maria Nadejde, Alexandra Birch, and Philipp Koehn. 2016b. A neural verb lexicon model with source-side syntactic context for string-to-tree machine translation. In *Proceedings of the International Workshop on Spoken Language Translation (IWSLT)*.

Maria Nadejde, Philip Williams, and Philipp Koehn. 2013. Edinburgh's Syntax-Based Machine Translation Systems. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 170–176, Sofia, Bulgaria.

Jan Niehues, Thanh-Le Ha, Eunah Cho, and Alex Waibel. 2016. Using factored word representation in neural network language models. In *Proceedings of the First Conference on Machine Translation*, Berlin, Germany.

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 on Association for Computational Linguistics*, ACL '02, pages 311–318, Stroudsburg, PA, USA. Association for Computational Linguistics.

Stefan Riezler and John T. Maxwell. 2005. On some pitfalls in automatic evaluation and significance testing for mt. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 57–64, Ann Arbor, Michigan. Association for Computational Linguistics.

Rico Sennrich. 2015. Modelling and Optimizing on Syntactic N-Grams for Statistical Machine Translation. *Transactions of the Association for Computational Linguistics*, 3:169–182.

Rico Sennrich, Orhan Firat, Kyunghyun Cho, Alexandra Birch, Barry Haddow, Julian Hitschler, Marcin Junczys-Dowmunt, Samuel Läubli, Antonio Valerio Miceli Barone, Jozef Mokry, and Maria Nadejde. 2017. Nematus: a toolkit for neural machine translation. In *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 65–68, Valencia, Spain. Association for Computational Linguistics.

Rico Sennrich and Barry Haddow. 2016. Linguistic input features improve neural machine translation. In *Proceedings of the First Conference on Machine Translation*, pages 83–91, Berlin, Germany.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Edinburgh neural machine translation systems for wmt 16. In *Proceedings of the First Conference on Machine Translation*, pages 371–376, Berlin, Germany. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Berlin, Germany. Association for Computational Linguistics.

Rico Sennrich, Martin Volk, and Gerold Schneider. 2013. Exploiting Synergies Between Open Re-

sources for German Dependency Parsing, POS-tagging, and Morphological Analysis. In *Proceedings of the International Conference Recent Advances in Natural Language Processing 2013*, pages 601–609, Hissar, Bulgaria.

Xing Shi, Inkit Padhi, and Kevin Knight. 2016. Does string-based neural mt learn source syntax? In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1526–1534, Austin, Texas. Association for Computational Linguistics.

Mark Steedman. 2000. *The syntactic process*, volume 24. MIT Press.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, NIPS'14, pages 3104–3112.

Philip Williams and Philipp Koehn. 2012. Ghkm rule extraction and scope-3 parsing in moses. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 388–394.