Multi-Source Syntactic Neural Machine Translation

Anna Currey
University of Edinburgh
a.currey@sms.ed.ac.uk

Kenneth Heafield
University of Edinburgh
kheafiel@inf.ed.ac.uk

Abstract

We introduce a novel multi-source technique for incorporating source syntax into neural machine translation using linearized parses. This is achieved by employing separate encoders for the sequential and parsed versions of the same source sentence; the resulting representations are then combined using a hierarchical attention mechanism. The proposed model improves over both seq2seq and parsed baselines by over 1 BLEU on the WMT17 English→German task. Further analysis shows that our multi-source syntactic model is able to translate successfully without any parsed input, unlike standard parsed methods. In addition, performance does not deteriorate as much on long sentences as for the baselines.

1 Introduction

Neural machine translation (NMT) typically makes use of a recurrent neural network (RNN) -based encoder and decoder, along with an attention mechanism (Bahdanau et al., 2015; Cho et al., 2014; Kalchbrenner and Blunsom, 2013; Sutskever et al., 2014). However, it has been shown that RNNs require some supervision to learn syntax (Bentivogli et al., 2016; Linzen et al., 2016; Shi et al., 2016). Therefore, explicitly incorporating syntactic information into NMT has the potential to improve performance. This is particularly true for source syntax, which can improve the model's representation of the source language.

Recently, there have been a number of proposals for using linearized representations of parses within standard NMT (Aharoni and Goldberg, 2017; Li et al., 2017; Nadejde et al., 2017). Linearized parses are advantageous because they can inject syntactic information into the models without significant changes to the architecture. However, using linearized parses in a sequence-to-sequence (seq2seq) framework creates some challenges, particularly when using source parses. First, the parsed sequences are significantly longer than standard sentences, since they contain node labels as well as words. Second, these systems often fail when the source sentence is not parsed. This can be a problem for inference, since the external parser may fail on an input sentence at test time. We propose a method for incorporating linearized source parses into NMT that addresses these challenges by taking both the sequential source sentence and its linearized parse simultaneously as input in a multi-source framework. Thus, the model is able to use the syntactic information encoded in the parse while falling back to the sequential sentence when necessary. Our proposed model improves over both standard and parsed NMT baselines.

2 Related Work

2.1 Seq2seq Neural Parsing

Using linearized parse trees within sequential frameworks was first done in the context of neural parsing. Vinyals et al. (2015) parsed using an attentional seq2seq model; they used linearized, unlexicalized parse trees on the target side and sentences on the source side. In addition, as in this work, they used an external parser to create synthetic parsed training data, resulting in improved parsing performance. Choe and Charniak (2016) adopted a similar strategy, using linearized parses in an RNN language modeling framework.

2.2 NMT with Source Syntax

Among the first proposals for using source syntax in NMT was that of Luong et al. (2016), who introduced a multi-task system in which the source data was parsed and translated using a shared encoder and two decoders. More radical changes to

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2961–2966
Brussels, Belgium, October 31 - November 4, 2018. ©2018 Association for Computational Linguistics

the standard NMT paradigm have also been proposed. Eriguchi et al. (2016) introduced tree-to-sequence NMT; this model took parse trees as input using a tree-LSTM (Tai et al., 2015) encoder. Bastings et al. (2017) used a graph convolutional encoder in order to take labeled dependency parses of the source sentences into account. Hashimoto and Tsuruoka (2017) added a latent graph parser to the encoder, allowing it to learn soft dependency parses while simultaneously learning to translate.

2.3 Linearized Parse Trees in NMT

The idea of incorporating linearized parses into seq2seq has been adapted to NMT as a means of injecting syntax. Aharoni and Goldberg (2017) first did this by parsing the target side of the training data and training the system to generate parsed translations of the source input; this is the inverse of our parse2seq baseline. Similarly, Nadejde et al. (2017) interleaved CCG supertags with words on the target side, finding that this improved translation despite requiring longer sequences.

Most similar to our multi-source model is the parallel RNN model proposed by Li et al. (2017). Like multi-source, the parallel RNN used two encoders, one for words and the other for syntax. However, they combined these representations at the word level, whereas we combine them on the sentence level. Their mixed RNN model is also similar to our parse2seq baseline, although the mixed RNN decoder attended only to words. As the mixed RNN model outperformed the parallel RNN model, we do not attempt to compare our model to parallel RNN. These models are similar to ours in that they incorporate linearized parses into NMT; here, we utilize a multi-source framework.

2.4 Multi-Source NMT

Multi-source methods in neural machine translation were first introduced by Zoph and Knight (2016) for multilingual translation. They used one encoder per source language, and combined the resulting sentence representations before feeding them into the decoder. Firat et al. (2016) expanded on this by creating a multilingual NMT system with multiple encoders and decoders. Libovický and Helcl (2017) applied multi-source NMT to multimodal translation and automatic post-editing and explored different strategies for combining attention over the two sources. In this paper, we apply the multi-source framework to a novel task, syntactic neural machine translation.

3 NMT with Linearized Source Parses

We propose a multi-source method for incorporating source syntax into NMT. This method makes use of linearized source parses; we describe these parses in section 3.1. Throughout this paper, we refer to standard sentences that do not contain any explicit syntactic information as *sequential*; see Table 1 for an example.

3.1 Linearized Source Parses

We use an off-the-shelf parser, in this case Stanford CoreNLP (Manning et al., 2014), to create binary constituency parses. These parses are linearized as shown in Table 1. We tokenize the opening parentheses with the node label (so each node label begins with a parenthesis) but keep the closing parentheses separate from the words they follow. For our task, the parser failed on one training sentence of 5.9 million, which we discarded, and succeeded on all test sentences. It took roughly 16 hours to parse the 5.9 million training sentences.

Following Sennrich et al. (2016b), our networks operate at the subword level using byte pair encoding (BPE) with a shared vocabulary on the source and target sides. However, the parser operates at the word level. Therefore, we parse then break into subwords, so a leaf may have multiple tokens without internal structure.

The proposed method is tested using both *lexicalized* and *unlexicalized* parses. In *unlexicalized* parses, we remove the words, keeping only the node labels and the parentheses. In *lexicalized* parses, the words are included. Table 1 shows an example of the three source sentence formats: sequential, lexicalized parse, and unlexicalized parse. Note that the lexicalized parse is significantly longer than the other versions.

3.2 Multi-Source

We propose a multi-source framework for injecting linearized source parses into NMT. This model consists of two identical RNN encoders with no shared parameters, as well as a standard RNN decoder. For each target sentence, two versions of the source sentence are used: the sequential (standard) version and the linearized parse (lexicalized or unlexicalized). Each of these is encoded simul-

	Example Sentence
sequential	history is a great teacher .
lexicalized parse	(ROOT (S (NP (NN history)) (VP (VBZ is) (NP (DT a) (JJ great) (NN teacher))) (. .)))
unlexicalized parse	(ROOT (S (NP (NN)) (VP (VBZ) (NP (DT) (JJ) (NN))) (. .)))
target sentence	die Geschichte ist ein großartiger Lehrmeister .

Table 1: Example source training sentence with sequential, lexicalized parse, and unlexicalized parse versions. We include the corresponding target sentence for reference.

taneously using the encoders; the encodings are then combined and used as input to the decoder. We combine the source encodings using the hierarchical attention combination proposed by Libovický and Helcl (2017). This consists of a separate attention mechanism for each encoder; these are then combined using an additional attention mechanism over the two separate context vectors. This multi-source method is thus able to combine the advantages of both standard RNN-based encodings and syntactic encodings.

4 Experimental Setup

4.1 Data

We base our experiments on the WMT17 (Bojar et al., 2017) English (EN) → German (DE) news translation task. All 5.9 million parallel training sentences are used, but no monolingual data. Validation is done on newstest2015, while newstest2016 and newstest2017 are used for testing.

We train a shared BPE vocabulary with 60k merge operations on the parallel training data. For the parsed data, we break words into subwords after applying the Stanford parser. We tokenize and truecase the data using the Moses tokenizer and truecaser (Koehn et al., 2007).

4.2 Implementation

The models are implemented in Neural Monkey (Helcl and Libovický, 2017). They are trained using Adam (Kingma and Ba, 2015) and have minibatch size 40, RNN size 512, and dropout probability 0.2 (Gal and Ghahramani, 2016). We train to convergence on the validation set, using BLEU (Papineni et al., 2002) as the metric.

For sequential inputs and outputs, the maximum sentence length is 50 subwords. For parsed inputs, we increase maximum sentence length to 150 subwords to account for the increased length due to the parsing labels; we still use a maximum output length of 50 subwords for these systems.

	System	**2016**	**2017**
baseline	seq2seq	25.0	20.8
	parse2seq	25.4	20.9
proposed	multi-source lex	**26.5**	**21.9**
	multi-source unlex	26.4	21.7

Table 2: BLEU scores on newstest2016 and newstest2017 datasets for the baselines, unlexicalized (unlex), and lexicalized (lex) systems.

4.3 Baselines

Seq2seq

The proposed models are compared against two baselines. The first, referred to here as *seq2seq*, is the standard RNN-based neural machine translation system with attention (Bahdanau et al., 2015). This baseline does not use the parsed data.

Parse2seq

The second baseline we consider is a slight modification of the mixed RNN model proposed by Li et al. (2017). This uses an identical architecture to the seq2seq baseline (except for a longer maximum sentence length in the encoder). Instead of using sequential data on the source side, the linearized parses are used. We allow the system to attend equally to words and node labels on the source side, rather than restricting the attention to words. We refer to this baseline as *parse2seq*.

5 Results

Table 2 shows the performance on EN→DE translation for each of the proposed systems and the baselines, as approximated by BLEU score.

The multi-source systems improve strongly over both baselines, with improvements of up to 1.5 BLEU over the seq2seq baseline and up to 1.1 BLEU over the parse2seq baseline. In addition, the lexicalized multi-source systems yields slightly higher BLEU scores than the unlexicalized multi-source systems; this is surprising because the lexicalized systems have significantly longer sequences than the unlexicalized ones. Finally, it is interesting to compare the seq2seq and parse2seq baselines. Parse2seq outperforms

System	Source Data	2016	2017
parse2seq	seq	0.6	0.5
multi-source lex	seq + seq	23.6	20.0
	seq + null	23.1	19.3
multi-source unlex	seq + seq	**23.7**	19.9
	seq + null	23.6	**20.9**

Table 3: BLEU scores on newstest2016 and newstest2017 when no parsed data is used during inference.

seq2seq by only a small amount compared to multi-source; thus, while adding syntax to NMT can be helpful, some ways of doing so are more effective than others.

6 Analysis

6.1 Inference Without Parsed Sentences

The parse2seq and multi-source systems require parsed source data at inference time. However, the parser may fail on an input sentence. Therefore, we examine how well these systems do when given only unparsed source sentences at test time.

Table 3 displays the results of these experiments. For the parse2seq baseline, we use only sequential (*seq*) data as input. For the lexicalized and unlexicalized multi-source systems, two options are considered: *seq* + *seq* uses identical sequential data as input to both encoders, while *seq* + *null* uses null input for the parsed encoder, where every source sentence is “( )”.

The parse2seq system fails when given only sequential source data. On the other hand, both multi-source systems perform reasonably well without parsed data, although the BLEU scores are worse than multi-source with parsed data.

6.2 BLEU by Sentence Length

For models that use source-side linearized parses (multi-source and parse2seq), the source sequences are significantly longer than for the seq2seq baseline. Since NMT already performs relatively poorly on long sentences (Bahdanau et al., 2015), adding linearized source parses may exacerbate this issue. To detect whether this occurs, we calculate BLEU by sentence length.

We bucket the sentences in newstest2017 by source sentence length. We then compute BLEU scores for each bucket for the seq2seq and parse2seq baselines and the lexicalized multi-source system. The results are in Figure 1.

In line with previous work on NMT on long sentences (Bahdanau et al., 2015; Li et al., 2017), we see a significant deterioration in BLEU for longer sentences for all systems. In particular, although the parse2seq model outperformed the seq2seq model overall, it does worse than seq2seq for sentences containing more than 30 words. This indicates that parse2seq performance does indeed suffer due to its long sequences. On the other hand, the multi-source system outperforms the seq2seq baseline for all sentence lengths and does particularly well for sentences with over 50 words. This may be because the multi-source system has both sequential and parsed input, so it can rely more on sequential input for very long sentences.

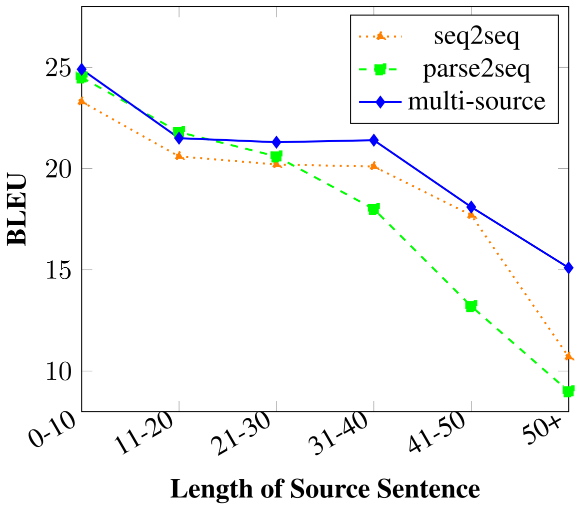

Figure 1: BLEU by sentence length on newstest2017 for baselines and lexicalized multi-source.

7 Conclusion

In this paper, we presented a multi-source method for effectively incorporating linearized parses of the source data into neural machine translation. This method, in which the parsed and sequential versions of the sentence were both taken as input during training and inference, resulted in gains of up to 1.5 BLEU on EN→DE translation. In addition, unlike parse2seq, the multi-source model translated reasonably well even when the source sentence was not parsed.

In the future, we will explore adding back-translated (Sennrich et al., 2016a) or copied (Currey et al., 2017) target data to our multi-source system. The multi-source model does not require all training data to be parsed; thus, monolingual data can be used even if the parser is unreliable for the synthetic or copied source sentences.

Acknowledgments

This work was funded by the Amazon Academic Research Awards program.

References

Roee Aharoni and Yoav Goldberg. 2017. Towards string-to-tree neural machine translation. In *Proceedings of the 55th Annual Meeting of the ACL*, pages 132–140. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations*.

Joost Bastings, Ivan Titov, Wilker Aziz, Diego Marcheggiani, and Khalil Sima'an. 2017. Graph convolutional encoders for syntax-aware neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1957–1967. Association for Computational Linguistics.

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*, pages 257–267. Association for Computational Linguistics.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Shujian Huang, Matthias Huck, Philipp Koehn, Qun Liu, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Raphael Rubino, Lucia Specia, and Marco Turchi. 2017. Findings of the 2017 Conference on Machine Translation (WMT17). In *Proceedings of the Second Conference on Machine Translation*, pages 169–214. Association for Computational Linguistics.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1724–1734. Association for Computational Linguistics.

Do Kook Choe and Eugene Charniak. 2016. Parsing as language modeling. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2331–2336. Association for Computational Linguistics.

Anna Currey, Antonio Valerio Miceli Barone, and Kenneth Heafield. 2017. Copied monolingual data improves low-resource neural machine translation. In *Proceedings of the Second Conference on Machine Translation*, pages 148–156. 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 ACL*, pages 823–833. Association for Computational Linguistics.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism. In *Proceedings of NAACL-HLT*, pages 866–875. Association for Computational Linguistics.

Yarin Gal and Zoubin Ghahramani. 2016. A theoretically grounded application of dropout in recurrent neural networks. In *Advances in Neural Information Processing Systems 29*, pages 1019–1027.

Kazuma Hashimoto and Yoshimasa Tsuruoka. 2017. Neural machine translation with source-side latent graph parsing. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 125–135. Association for Computational Linguistics.

Jindřich Helcl and Jindřich Libovickỳ. 2017. Neural Monkey: An open-source tool for sequence learning. *The Prague Bulletin of Mathematical Linguistics*, 107(1):5–17.

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. Association for Computational Linguistics.

Diederik Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL*, pages 177–180. Association for Computational Linguistics.

Junhui Li, Deyi Xiong, Zhaopeng Tu, Muhua Zhu, Min Zhang, and Guodong Zhou. 2017. Modeling source syntax for neural machine translation. In *Proceedings of the 55th Annual Meeting of the ACL*, pages 688–697. Association for Computational Linguistics.

Jindřich Libovickỳ and Jindřich Helcl. 2017. Attention strategies for multi-source sequence-to-sequence learning. In *Proceedings of the 55th Annual Meeting of the ACL*, pages 196–202. Association for Computational Linguistics.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016. Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535.

Minh-Thang Luong, Quoc V Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2016. Multi-task sequence to sequence learning. In *4th International Conference on Learning Representations*.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of the 52nd Annual Meeting of the ACL*, pages 55–60. Association for Computational Linguistics.

Maria Nadejde, Siva Reddy, Rico Sennrich, Tomasz Dwojak, Marcin Junczys-Dowmunt, Philipp Koehn, and Alexandra Birch. 2017. Predicting target language CCG supertags improves neural machine translation. In *Proceedings of the Second Conference on Machine Translation*, pages 68–79. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the ACL*, pages 311–318. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the ACL*, pages 86–96. 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 ACL*, pages 1715–1725. Association for Computational Linguistics.

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. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27*, pages 3104–3112.

Kai Sheng Tai, Richard Socher, and Christopher Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the ACL and the 7th International Joint Conference on Natural Language Processing*, pages 1556–1566. Association for Computational Linguistics.

Oriol Vinyals, Łukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, and Geoffrey Hinton. 2015. Grammar as a foreign language. In *Advances in Neural Information Processing Systems 28*, pages 2773–2781.

Barret Zoph and Kevin Knight. 2016. Multi-source neural translation. In *Proceedings of NAACL-HLT*, pages 30–34. Association for Computational Linguistics.