# Improving Neural Translation Models with Linguistic Factors

**Cong Duy Vu Hoang**
University of Melbourne
Melbourne, VIC, Australia
vhoang2@student.unimelb.edu.au

**Gholamreza Haffari**
Monash University
Clayton, VIC, Australia
gholamreza.haffari@monash.edu

**Trevor Cohn**
University of Melbourne
Melbourne, VIC, Australia
t.cohn@unimelb.edu.au

## Abstract

This paper presents an extension of neural machine translation (NMT) model to incorporate additional word-level linguistic factors. Adding such linguistic factors may be of great benefits to learning of NMT models, potentially reducing language ambiguity or alleviating data sparseness problem (Koehn and Hoang, 2007). We explore different linguistic annotations at the word level, including: lemmatization, word clusters, Part-of-Speech tags, and labeled dependency relations. We then propose different neural attention architectures to integrate these additional factors into the NMT framework. Evaluating on translating between English and German in two directions with a low resource setting in the domain of TED talks, we obtain promising results in terms of both perplexity reductions and improved BLEU scores over baseline methods.

## 1 Introduction

Neural Machine Translation (NMT) (Devlin et al., 2014; Bahdanau et al., 2015) is a new paradigm in machine translation (MT) powered by recent advances in sequence to sequence learning frameworks (Graves, 2013; Sutskever et al., 2014). NMT has already made remarkable results and improvements over conventional SMT (Luong et al., 2015).

The core idea of NMT is the encoder-decoder framework where an *encoder* encodes the source sequence into a vector representation, and then a *decoder* generates the target sequence sequentially via a recurrent neural network (RNN). The use of a RNN provides the ability to memorize longer range dependencies that are impossible with standard $n$-gram modeling - a core component of the traditional Statistical Machine Translation (SMT) framework (Koehn et al., 2003; Lopez, 2008; Koehn, 2010). Unlike the traditional SMT, NMT offers unique mechanisms to learn translation equivalence without extensive feature engineering efforts.

Though promising, NMT still lacks of the ability of modeling deeper semantic and syntactic aspects of the language. Koehn and Hoang (2007) presented a *factored* translation model to address this issue for the traditional SMT framework (Koehn et al., 2007), where the model incorporates various linguistic annotations for the surface level words. Particularly for low-resource conditions, these extra annotations can lead to better translation of OOVs (or low-count words) and resolve ambiguities, hence increase the generalization capabilities of the model.

In machine translation with a low-resource setting, resolving data sparseness and semantic ambiguity problems can help improve its performance. In this paper, we investigate utilizing extra syntactic and semantic linguistic factors in the context of the NMT framework. Linguistic factors can include bundles of features, e.g., stems, roots, lemmas, morphological classes, data-driven clusters, syntactic analyses (part-of-speeches, constituency parsing, dependency parsing). Adding such extra factors may be of great benefits to NMT models, potentially reducing language ambiguity and alleviating data sparseness further. In this paper, we explore four word-level factor annotations, including: lemmatization, word clusters, Part-of-Speech tags, and relation labels in dependency parse trees (see Figure 1 for an example). We then propose different neural attention architec-

Cong Duy Vu Hoang, Reza Haffari and Trevor Cohn. 2016. Improving Neural Translation Models with Linguistic Factors.
In *Proceedings of Australasian Language Technology Association Workshop*, pages 7–14.

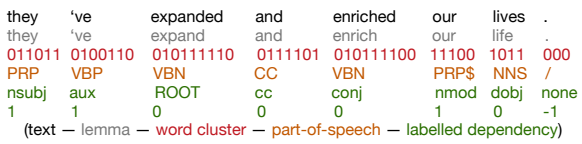



Figure 1: An example of linguistic factor annotations for a source sentence in English.

tures to integrate these additional factors into the NMT framework. Evaluating on translating between English and German in two directions with a low resource setting in the TED talks data, we obtain perplexity reductions and improved BLEU score over the baseline.

## 2 Incorporating Linguistic Factors

In this work, we investigate the feasibility of factored model idea (Koehn and Hoang, 2007) into attentional neural translation model (Bahdanau et al., 2015). As an initial work, we aim to find how the neural model can benefit from incorporating the additional linguistic factors in source language. Our work is an extension of (Bahdanau et al., 2015) with the integration of additional linguistic factors. A fully factored neural translation model for both source and target sides is considered as our future work. The following section will discuss our extensions of (Bahdanau et al., 2015) in §2.1. Assume that we have $L$ layers of linguistic factor annotations. The training data then consists of $N$ training parallel sentences $\{(\{\boldsymbol{x}^{(n,\ell)}\}_{\ell=0}^{L}, \boldsymbol{y}^{(n)})\}_{n=1}^{N}$ where the word sequence of the $n$th sentence-pair is denoted in the layer zero $\boldsymbol{x}^{(n,0)}$, its length is denoted by $|\boldsymbol{x}^{(n)}|$, its $L$ layers of annotations are denoted by $\{\boldsymbol{x}^{(n,\ell)}\}_{\ell=1}^{L}$, and the target sentence is denoted by $\boldsymbol{y}^{(n)}$. In what follows, we review and extend the attentional encoder-decoder neural machine translation for this setting, and explore various neural attention mechanisms operating on the multiple layers of linguistic factors over the source sentence.

### 2.1 Multi-Factor Encoder-Decoder

**Encoder.** First, to encode the source-side information, we first run each layer of linguistic annotations through bidirectional RNNs (biRNN) for dynamically representing the sequence embeddings, i.e.,

$$\boldsymbol{h}_j^{\ell} = \text{biRNN}_{enc}^{\ell,\psi}\left(\boldsymbol{x}_j^{\ell}, \left[\overrightarrow{\boldsymbol{h}^{\ell}}_{j-1}; \overleftarrow{\boldsymbol{h}^{\ell}}_{j+1}\right]^T\right); \quad (1)$$

where $\boldsymbol{x}_j^{\ell} \in \mathbb{R}^{H^{\ell}}$ is the word embedding at position $j$ in sequence layer $\ell$, and $\overrightarrow{\boldsymbol{h}^{\ell}}_j$ and $\overleftarrow{\boldsymbol{h}^{\ell}}_j$ are the RNN[1] hidden states. This encoding scheme captures not only the position specific information, but also the information coming from the left and right contexts.

**Decoder.** Next, a *decoder* operated by another RNN is used to predict the target $\boldsymbol{y}$ sequentially, from left to right:

$$\boldsymbol{g}_i = \text{RNN}_{dec}^{\phi}\left(\boldsymbol{c}_i, y_{i-1}, \boldsymbol{g}_{i-1}\right)$$
$$y_i \sim \text{softmax}\left(\boldsymbol{W}_o \cdot \text{MLP}\left(\boldsymbol{c}_i, y_{i-1}, \boldsymbol{g}_i\right) + \boldsymbol{b}_o\right);$$

where MLP is a single hidden layer neural network with $\tanh$ activation. The model parameters include $\phi$ the weight matrix $W_o \in \mathbb{R}^{V_y \times H}$ and the bias $\boldsymbol{b}_o \in \mathbb{R}^{V_y}$, with $V_y$ and $H$ denoting the target vocabulary size and hidden dimension size, respectively.

Note that the state of the decoder $\boldsymbol{g}_i$ is conditioned on its previous state $\boldsymbol{g}_{i-1}$, the previously generated target word $\boldsymbol{y}_{i-1}$, and the source side *context* $\boldsymbol{c}_i$ summarizing the areas of the source sentence needs to be *attended* to. Finally, the model is trained end-to-end by minimizing the cross-entropy loss over the target sequence and stochastic gradient descent (SGD) is used for optimizing the model parameters .

In what follows, we explore various attention mechanisms for our case where the input sentence is annotated with multiple linguistic factors, and show how the source context $\boldsymbol{c}_i$ is constructed.

### 2.2 Multi-Factor Attention Architectures

In this paper, we explore various attention mechanisms of integrating linguistic factors as briefly summarized in Figure 2, including Global Attention, Local Attention, and hybrid Global-Local Attention.

**Global Attention.** Our first approach has one shared attention vector for all the annotation layers, forcing each layer to attend to the same positions. This essentially means stacking the representations of all the input embeddings $\boldsymbol{x}^{\ell}$ into one vector, i.e., $\boldsymbol{x}_j^g = \left[\boldsymbol{x}_j^0, \ldots, \boldsymbol{x}_j^L\right]^T$. This stacked

[1] Generally, an RNN can be employed as Long-Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) or Gated Recurrent Unit (GRU) (Cho et al., 2014). Since the RNN recurrent structure is not our focus, we ignored its formulation in this paper.

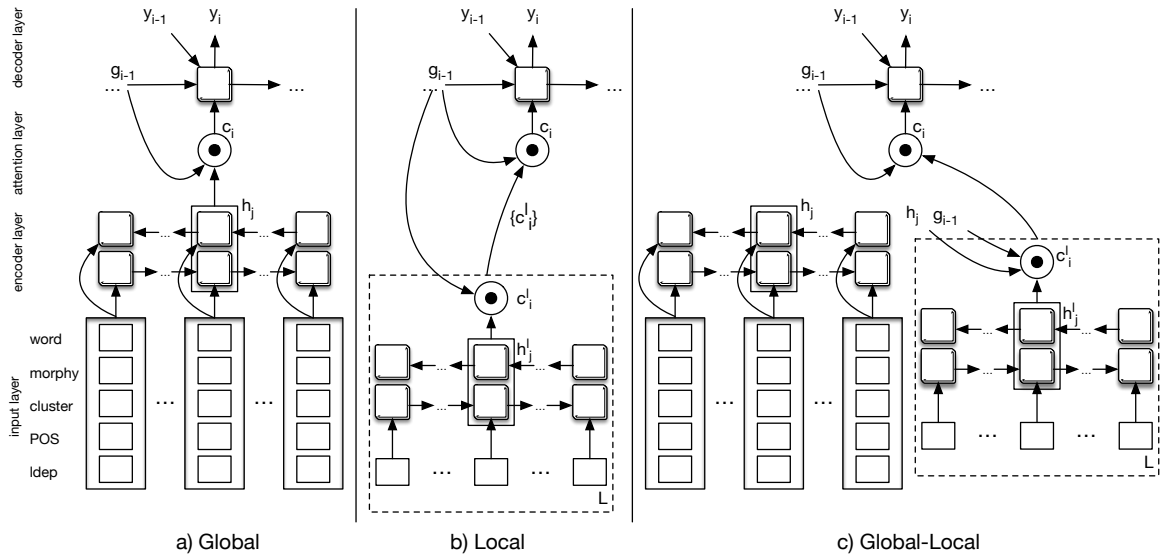



Figure 2: Proposed attention architectures of integrating linguistic factors for the NMT framework.

representation is used in place of only word embedding $\boldsymbol{x}_j$ to encode the input position (eqn 1) to $\boldsymbol{h}^g_j$. It is then used to construct the source context for the decoder, using $\boldsymbol{c}_i = \sum_{j=1}^{|\boldsymbol{x}|} \alpha_{ij} \boldsymbol{h}^g_j$ with

$$\boldsymbol{\alpha}_i = \text{softmax}(\boldsymbol{e}_i) \quad ; \quad e_{ij} = \text{MLP}\left(\boldsymbol{g}_{i-1}, \boldsymbol{h}^g_j\right)$$

$$\boldsymbol{h}^g_j = \text{biRNN}^{\theta}_{enc}\left(\boldsymbol{x}^g_j, \left[\overrightarrow{\boldsymbol{h}^g}_{j-1}; \overleftarrow{\boldsymbol{h}^g}_{j+1}\right]^T\right),$$

where scalar $e_{ij}$ denotes the unnormalized alignment probability between the source word annotation $j$ and target word $i$, which is produced by single hidden layer neural network with $\tanh$ activation.

**Local Attention.** The model may benefit from different attentions learned for different layers. Thus, the second idea is to have multiple attentions for linguistic layers independently, and compute layer-specific context vectors $\{\boldsymbol{c}^\ell_i\}^L_{\ell=0}$ and stack them up:

$$\boldsymbol{c}_i = \left[\boldsymbol{c}^0_i, \ldots, \boldsymbol{c}^L_i\right]^T \quad ; \quad \boldsymbol{c}^\ell_i = \sum_{j=1}^{T_x} \alpha^\ell_{ij} \boldsymbol{h}^\ell_j$$

$$\boldsymbol{\alpha}^\ell_i = \text{softmax}(\boldsymbol{e}^\ell_i) \quad ; \quad e^\ell_{ij} = \text{MLP}\left(\boldsymbol{g}_{i-1}; \boldsymbol{h}^\ell_j\right)$$

where $e^\ell_{ij}$ denotes the alignment score between the annotation at layer $\ell$ and the target word. The MLP for each layer has a different parameterization.

**Global-Local Attention.** Finally, we consider a hybrid global-local attention mechanism which makes use of the *global* hidden representation $\boldsymbol{h}^g$ across all of the layers in generating the *local* attentions, formulated as:

$$e^\ell_{ij} = \text{MLP}\left(\boldsymbol{g}_{i-1}, \boldsymbol{h}^g_j\right).$$

In contrast to the local attention the attention for layer $\ell$ depends on the global encoding, $\boldsymbol{h}^g$, rather than the local encoding for that layer, $\boldsymbol{h}^l$.

In training, we *encourage* the model to have similar attentions across the layers by adding a penalty term to the cross-entropy training objective,

$$\sum_{n=1}^{N} \sum_{i=1}^{|\boldsymbol{y}^{(n)}|} \sum_{\ell=0}^{L} \left\| \bar{\boldsymbol{\alpha}}^{(n)}_i - \boldsymbol{\alpha}^{(n),\ell}_i \right\|^2_2$$

where $\boldsymbol{\alpha}^{(n),\ell}_i$ is the attention to the layer $\ell$ when generating the target word $i$, and we define $\bar{\boldsymbol{\alpha}}^{(n)}_i := \frac{1}{L+1} \sum^L_{\ell=0} \boldsymbol{\alpha}^{(n),\ell}_i$ as the average attention across all layers. Essentially, our regularizer penalizes parameters which induce layer-specific attentions deviating from the average attention.

## 3 Experiments

**Data.** We conducted our experiments on TED Talks datasets (Cettolo et al., 2012) and translate between English (en) $\leftrightarrow$ German (de). For training, we used about 200K parallel sentences, and used tst2010 for tuning model parameters (phrase-based SMT) and early stopping (NMT). We evaluated on the official test sets tst2013 and tst2014,

| dataset | # tokens (K) | | # types (K) | | # sents | # docs |
|---|---|---|---|---|---|---|
| **en↔de** | | | | | | |
| train | 4384.68 | 4161.58 | 19.42 | 26.22 | 198968 | 1597 |
| tune-tst2010 | 35.13 | 33.42 | 3.29 | 3.87 | 1565 | 16 |
| test1-tst2013 | 22.86 | 21.64 | 2.67 | 3.08 | 993 | 15 |
| test2-tst2014 | 27.40 | 26.44 | 3.21 | 3.66 | 1305 | 16 |

Table 1: Statistics of the training & evaluation sets from IWSLT'14,15 MT track (including en↔de) showing in each cell the count for the source language (left) and target language (right). "#types" refers to filtered vocabulary with word frequency cut-off 5.

| configuration | tst2013 | tst2014 | #param (M) |
|---|---|---|---|
| **en→de** | | | |
| Vanilla Attentional Model | 8.20 | 10.98 | 47.80 |
| w/ *glo*+all-factors | **7.84** | **10.35** | 50.88 |
| w/ *loc*+all-factors | 8.02 | 10.80 | 52.06 |
| w/ *glo-loc*+all-factors (w/o regularization penalty) | **7.81** | **10.28** | 57.52 |
| w/ *glo-loc*+all-factors (w/ regularization penalty) | **7.48**♠ | **10.15**♠ | 57.52 |
| **de→en** | | | |
| Vanilla Attentional Model | 8.76 | 11.81 | 44.46 |
| w/ *glo*+all-factors | **8.50** | **11.26** | 47.58 |
| w/ *loc*+all-factors | **8.50** | **11.48** | 48.76 |
| w/ *glo-loc*+all-factors (w/ regularization penalty) | **8.29**♠ | **10.95**♠ | 54.22 |

Table 2: Perplexity scores for attentional model variants evaluated on en↔de translations, and "#param" refers to no. of model parameters (in millions). **bold**: "statistically significantly better than vanilla attentional model", ♠: best performance.

following Cettolo et al. (2014). We chose a word frequency cut-off of $\geq 5$ for limiting the vocabulary when training neural models, resulting in 19K and 26K word types for English and German, respectively. All details of data statistics can be found in Table 1.

As linguistic factors, we annotated the source sentences with lemmas,[2] word clusters,[3] and POS tags. We also annotated with the labelled dependency, i.e. by taking the dependency label between each word and its head (together with its direction, i.e. left or right)[4] in the dependency parse tree. Also note that the POS tags and dependency parse trees were extracted from parsing results produced by Stanford Parser[5] and ParZu.[6]

**Set-up and Baselines.** We used the *cnn* library[7] for our implementation. All neural models were configured with 512 input embedding and hidden layer dimensions, and 384 alignment dimension, with 1 and 2 hidden layers in the source and target, respectively. We employed LSTM recurrent structure (Hochreiter and Schmidhuber, 1997) for both source and target RNN sequences. For the phrase-based SMT baseline, we used the Moses toolkit (Koehn et al., 2007) with its standard configuration. To encode the linguistic factors, we used 128, 64, 64, 64 embedding dimensions for each of lemma, word cluster, Part-of-Speech (POS), and labelled dependency sequences, respectively. For training our neural models, the best perplexity scores on tuning sets were used for early stopping of training, which was usually between 5-8 epochs. For decoding, we used a simple greedy algorithm with length normalization. For evaluation of translations, we applied bootstrapping resampling (Koehn, 2004) to measure the statistical significance ($p < 0.05$) of BLEU score differences between translation outputs of proposed models compared to the baselines.

**Results and Analysis.** We report our experimental results based on standard perplexity and BLEU (Papineni et al., 2002) scores, as shown in Tables 2 and 3, respectively. Table 2 shows that the attentional model with our extensions is noticeably better than the vanilla NMT in terms of perplexity. Among the three attention architectures,

[2]NLTK, `http://www.nltk.org/`

[3]Brown clustering, `https://github.com/percyliang/brown-cluster`

[4]The direction is encoded effectively as 3-bit vector.

[5]`http://nlp.stanford.edu/software/lex-parser.shtml` (en)

[6]`https://github.com/rsennrich/ParZu` (de)

[7]`https://github.com/clab/cnn/tree/master/cnn`

| configuration | tst2013 | tst2014 |
|---|---|---|
| **en→de** | | |
| Moses baseline | 21.31 | 19.16 |
| Vanilla Attentional Model | 25.03 | 20.96 |
| w/ *glo*+all-factors | **25.43** | **22.15**♠ |
| w/ *loc*+all-factors | 25.04 | 21.24 |
| w/ *glo-loc*+all-factors (w/o regularization penalty) | 25.06 | 21.29 |
| w/ *glo-loc*+all-factors (w/ regularization penalty) | **25.92**♠ | **21.84** |
| **de→en** | | |
| Moses baseline | 29.96 | 25.13 |
| Vanilla Attentional Model | 29.85 | 24.84 |
| w/ *glo*+all-factors | 29.63 | **25.30**♠ |
| w/ *loc*+all-factors | 29.32 | 24.40 |
| w/ *glo-loc*+all-factors (w/ regularization penalty) | **30.45**♠ | 24.72 |

Table 3: BLEU scores for attentional model variants evaluated on en↔de translations.

the *glo-loc* attention outperformed others, giving significant improvement compared to the vanilla model. The use of the *loc* attention did not give much improvement. We suspect that the learned model itself has difficulties deciding which factors to attend to. The drawback of the *glo* attention is that it enforces only one attention mechanism for all of the layers. This may cause the loss of individual effects that potentially exist in each of layers. The *glo-loc* attention aims at taking advantage of *glo* attention and solving the limitation of *loc* attention with the penalty term, hence giving better performance.

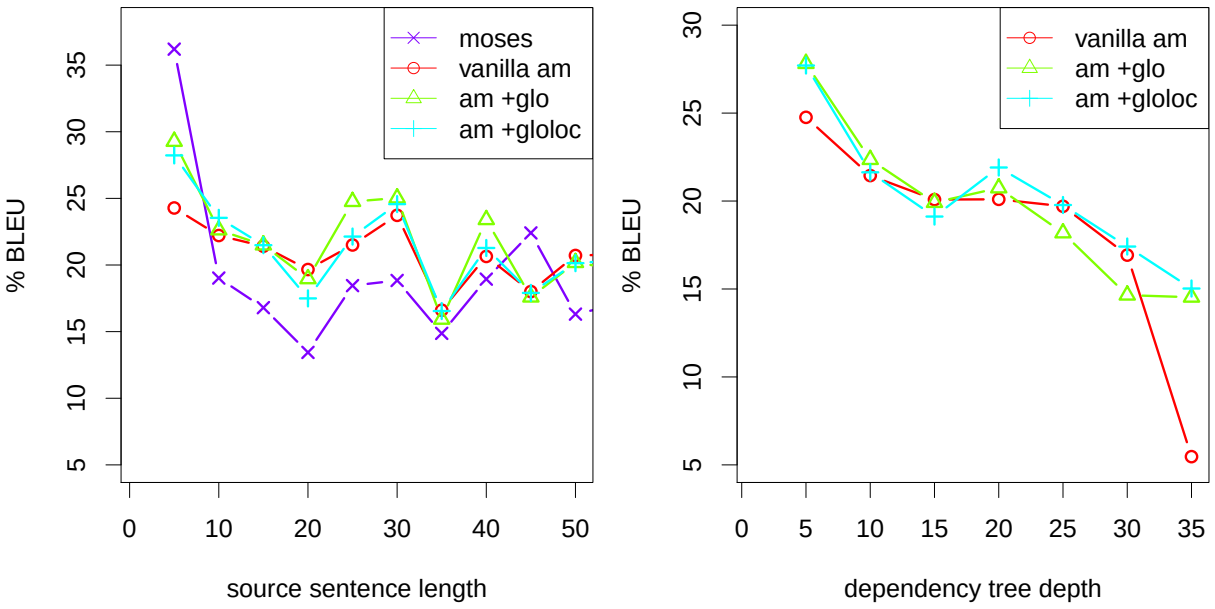
Table 3 shows the BLEU score results. Compared to Moses baseline, the vanilla attentional model is superior for en→de and comparable for de→en translation tasks. It is noticeable that the attentional model is capable of working remarkably well, despite the relatively small amounts of parallel data. However, table 3 shows the inconsistency, compared to the respective perplexity scores in Table 2. For en→de, both *glo* and *glo-loc* attention architectures worked competitively well, giving significantly better BLEU scores than the vanilla attentional model. Compared to *glo*, the *glo-loc* attention is superior in tst2013, but slightly detrimental in tst2014 although (its respective perplexity scores are better). These results show that reductions in perplexity scores do not guarantee improved BLEU scores, which is particularly true for de→en translation.

For the analysis, we further investigate the improvement of the translation quality versus sentence complexity. This would show the extent to which the extra linguistic layers have been helpful in resolving ambiguities of source sentences in translation. We formalize sentence complexity by taking either its length or the depth of its parse tree into consideration. Figure 3 and 4 plot the BLEU score versus these two measures of complexity in two evaluation sets. As seen, the extra linguistic layers has helped the translation quality of more complex sentences compared to the vanilla attentional model.

## 4 Related Work

Recent advances in deep learning research facilitate innovative ideas in machine translation. The attentional encoder-decoder framework pioneered by Bahdanau et al. (2015) is the core, opening a new trend in neural machine translation. Luong et al. (2015) followed the work of (Bahdanau et al., 2015) by experimenting various options on the generation of soft alignments with global and local attention mechanisms. Inspired by remarkable characteristics of state-of-the-art SMT models, Cohn et al. (2016) incorporated structural alignment biases inspired from conventional statistical alignment models (e.g. IBM models 1, 2) to encourage more linguistic structures in the alignment process. Similar in spirit to this, Feng et al. (2016) made use of additional RNN structure for the attention mechanism, hence likely capturing long range dependencies between the attention vectors. Tu et al. (2016) further proposed a so-called coverage vector to trace the attention history for flexibly adjusting future attentions.

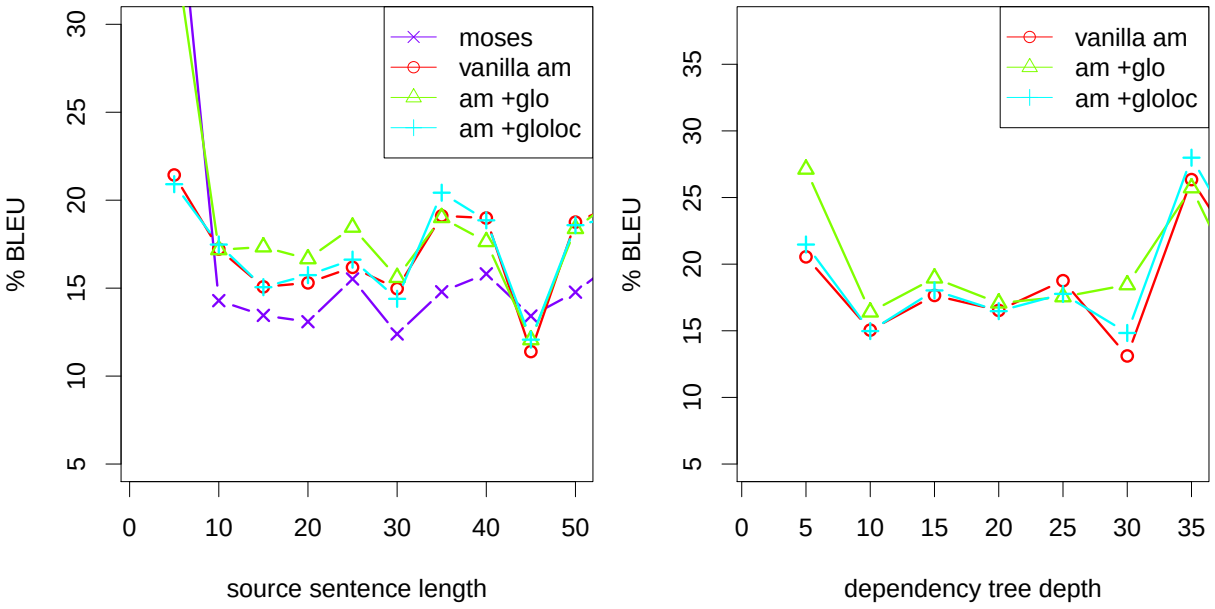
Though having been developed for almost 2 years, the NMT models are currently competitive with state-of-the-art SMT models. However, NMT models are still lacking of capabilities to modelling shallow language characteristics, e.g. the additional annotation at word level of linguistic factors. Such kinds of factors can provide extra




(a) source sentence length vs. BLEU

(b) dependency tree depth vs. BLEU

Figure 3: Analysis based on the evaluation set tst2013 in en→de translation.




(a) source sentence length vs. BLEU

(b) dependency tree depth vs. BLEU

Figure 4: Analysis based on the evaluation set tst2014 in en→de translation.

dimensions for data sparseness problem as shown in earlier works in SMT models, e.g., (Zhang and Sumita, 2007; Rishøj and Søgaard, 2011; Wuebker et al., 2013). The most closely related work to ours is the factored translation model for SMT framework proposed by Koehn and Hoang (2007). This model evaluated the effects of various linguistic factors (including lemma, POS, morphology) which are annotated for both source and target sides. Our work explored the same manner in the context of NMT framework though only considering source side. However, we further explored the annotation with labelled dependency which potentially inject syntactic information into neural model. Concurrent to our work, Sennrich and Haddow (2016) proposed similar idea for the NMT framework, however, their work has only explored the so-called global attention whereas we proposed more attention mechanisms with local and hybrid global-local attentions. Also, our ex-

periments were conducted in a low-resourced setting in a different domain with TED talk data.

## 5 Conclusion & Future Work

In this paper, we have presented a novel attentional encoder-decoder for translation capable of integrating linguistic factors in the source language. Four linguistic factors were evaluated, including lemmatization, word clustering, part-of-speech tagging, and labeled dependencies. We proposed several neural attention mechanisms operating over the factors. Our experimental results on two language pairs show that the neural translation model with integrated linguistic factors can be improved, in terms of both perplexity and BLEU scores.

As our future work, we aim to explore whether the attentional neural translation model can benefit from linguistic factors, operating over the *target* language. This work can be considered as the first work towards fully-factored neural translation model.

## References


Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *Proc. of 3rd International Conference on Learning Representations (ICLR2015)*.

Mauro Cettolo, Christian Girardi, and Marcello Federico. 2012. WIT$^3$: Web Inventory of Transcribed and Translated Talks. In *Proceedings of the 16$^{th}$ Conference of the European Association for Machine Translation (EAMT)*, pages 261–268, Trento, Italy, May.

M. Cettolo, J. Niehues, S. Stuker, L. Bentivogli, and M. Federico. 2014. Report on the 11th IWSLT Evaluation Campaign. In *Proc. of The International Workshop on Spoken Language Translation (IWSLT)*.

Kyunghyun Cho, Bart van Merrienboer, 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 (EMNLP)*, pages 1724–1734, Doha, Qatar, October. Association for Computational Linguistics.

T. Cohn, C. D. V. Hoang, E. Vymolova, K. Yao, C. Dyer, and G. Haffari. 2016. Incorporating Structural Alignment Biases into an Attentional Neural Translation Model. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, San Diego, California, June. Association for Computational Linguistics.

Jacob Devlin, Rabih Zbib, Zhongqiang Huang, Thomas Lamar, Richard Schwartz, and John Makhoul. 2014. Fast and Robust Neural Network Joint Models for Statistical Machine Translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1370–1380, Baltimore, Maryland, June. Association for Computational Linguistics.

S. Feng, S. Liu, M. Li, and M. Zhou. 2016. Implicit Distortion and Fertility Models for Attention-based Encoder-Decoder NMT Model. *ArXiv e-prints*, January.

A. Graves. 2013. Generating Sequences With Recurrent Neural Networks. *ArXiv e-prints*, August.

Sepp Hochreiter and Jurgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Comput.*, 9(8):1735–1780, November.

Philipp Koehn and Hieu Hoang. 2007. Factored Translation Models. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 868–876, Prague, Czech Republic, June. Association for Computational Linguistics.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical Phrase-based Translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 48–54, Stroudsburg, PA, USA. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 177–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

Philipp Koehn. 2004. Statistical Significance Tests for Machine Translation Evaluation. In Dekang Lin and Dekai Wu, editors, *Proceedings of Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pages 388–395, Barcelona, Spain, July. Association for Computational Linguistics.

Philipp Koehn. 2010. *Statistical Machine Translation*. Cambridge University Press, New York, NY, USA, 1st edition.

Adam Lopez. 2008. Statistical Machine Translation. *ACM Comput. Surv.*, 40(3):8:1–8:49, August.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective Approaches to Attention-based Neural Machine Translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal, September. 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 on Association for Computational Linguistics*, ACL '02, pages 311–318, Stroudsburg, PA, USA. Association for Computational Linguistics.

Christian Rishøj and Anders Søgaard. 2011. Factored Translation with Unsupervised Word Clusters. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, WMT '11, pages 447–451, Stroudsburg, PA, USA. Association for Computational Linguistics.

Rico Sennrich and Barry Haddow. 2016. Linguistic input features improve neural machine translation. In *Proc. of the First Conference on Machine Translation (WMT16)*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to Sequence Learning with Neural Networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 3104–3112. Curran Associates, Inc.

Z. Tu, Z. Lu, Y. Liu, X. Liu, and H. Li. 2016. Coverage-based Neural Machine Translation. In *Proceedings of the 4th International Conference on Learning Representations (ICLR 2016 Workshop Track)*, ICLR '16 Workshop Track.

Joern Wuebker, Stephan Peitz, Felix Rietig, and Hermann Ney. 2013. Improving Statistical Machine Translation with Word Class Models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1377–1381, Seattle, Washington, USA, October. Association for Computational Linguistics.

Ruiqiang Zhang and Eiichiro Sumita. 2007. Boosting Statistical Machine Translation by Lemmatization and Linear Interpolation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 181–184, Stroudsburg, PA, USA. Association for Computational Linguistics.