Tag Assisted Neural Machine Translation of Film Subtitles

Aren Siekmeier[*], **WonKee Lee**[*], **Hongseok Kwon**[†], and **Jong-Hyeok Lee**[*†]
Pohang University of Science and Technology
[*]Department of Computer Science and Engineering
[†]Graduate School of Artificial Intelligence
{asiekmeier,wklee,hkwon,jhlee}@postech.ac.kr

Abstract

We implemented a neural machine translation system that uses automatic sequence tagging to improve the quality of translation. Instead of operating on unannotated sentence pairs, our system uses pre-trained tagging systems to add linguistic features to source and target sentences. Our proposed neural architecture learns a combined embedding of tokens and tags in the encoder, and simultaneous token and tag prediction in the decoder. Compared to a baseline with unannotated training, this architecture increased the BLEU score of German to English film subtitle translation outputs by 1.61 points using named entity tags; however, the BLEU score decreased by 0.38 points using part-of-speech tags. This demonstrates that certain token-level tag outputs from off-the-shelf tagging systems can improve the output of neural translation systems using our combined embedding and simultaneous decoding extensions.

1 Introduction

Neural machine translation (NMT) uses neural networks to translate unannotated text between a source and target language, but without additional linguistic information certain ambiguous inputs may be translated incorrectly. Consider the following examples:

1) Titanic struggles between good and evil.
   ✓ 선과 악 사이의 엄청난 투쟁.
   big fight between good and evil
   ✗ 타이타닉은 선과 악 사이에서 투쟁 중이다.
   The Titanic is fighting between good and evil

2) Titanic struggles to stay afloat.
   ✓ 타이타닉은 침몰하지 않도록 고군분투 중이다.
   The Titanic is struggling not to sink
   ✗ 침몰하지 않기 위한 엄청난 투쟁.
   big fight not to sink

In (1), "Titanic" is best translated as a common adjective; in (2), it most likely refers to a named entity, the famous ship. In addition to the bare token sequences, part-of-speech or named entity annotation of each token, provided manually or automatically, could provide additional information to improve the quality of translation.

Natural language processing (NLP) tools have benefited from the same explosion in deep learning and neural network developments that has spurred NMT. NLP tools include part-of-speech (POS) taggers, identifying the syntactic function of each input token, and named entity recognition systems. Named entity recognition (NER) identifies which tokens refer to named entities, including proper nouns such as people, place names, organizations, or dates. Recently, automatic named entity recognition (NER) systems have seen much development and refinement with the same deep learning tools used for NMT (Li et al., 2020). Automatic neural NER systems have achieved accuracy exceeding 92% F_1 scores in many languages and domains (Wang et al., 2019; Akbik et al., 2018). NER tags produced by these systems are useful in many other natural language processing contexts, such as coreference resolution, entity linking, or entity extraction (Ferreira Cruz et al., 2020). POS taggers have also achieved very high accuracy exceeding 98% on public treebank datasets (Akbik et al., 2018). We aim to use tags from publicly available pre-trained tagging systems as additional features to improve NMT training and output.

Tag assisted NMT requires modifications to the neural architecture to accommodate a tag at each token position. The encoder must learn an embedding that combines information from each token and its tag, then compute a hidden state from these embeddings. The decoder must learn to predict tokens and their tags simultaneously from the decoder state. Adding tag information to the predic-

Proceedings of the 18th International Conference on Spoken Language Translation, pages 255–262
Bangkok, Thailand (Online), August 5–6, 2021.

tion and corresponding training loss encourages the model to incorporate this information into its latent representations to improve outputs.

Compared to an untagged baseline system on word-tokenized data, our tagged translation system improved the BLEU score by 1.61 points on German to English parallel film subtitles data tagged with publicly available pre-trained named entity recognition systems, while part-of-speech tagging decreased the score by 0.38 BLEU points. Subword tokenization reduced these effects to +0.22 points and –0.22 points respectively. Nonetheless, this demonstrates the feasibility of using certain pre-trained tagging outputs to improve translation quality.

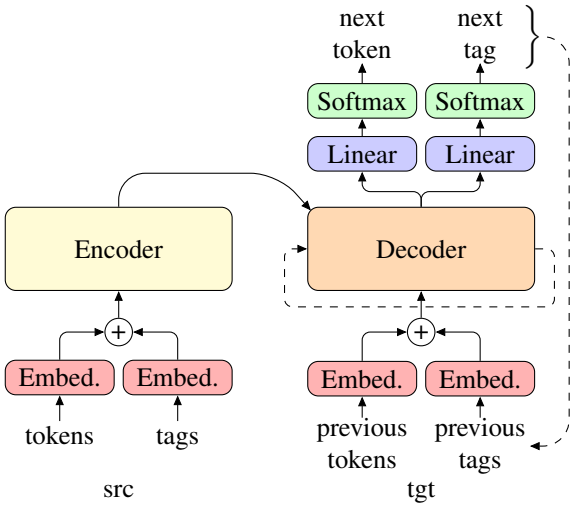

Figure 1: Tagged seq2seq

2 Related Work

Very early work addressed named entity translation by treating automatically identified named entities with a special translation system, usually a transliterator (Babych and Hartley, 2003). This work did not attempt to integrate the translation models for one to benefit from information learned by the other.

Later, especially with neural machine translation (NMT) systems, source-side feature augmentation research studied the inclusion of linguistic feature information into the source-side token embeddings, usually by adding in or concatenating additional learned feature vectors to the token embedding vectors, as we do in this work (Sennrich and Haddow, 2016; Hoang et al., 2016b; Ugawa et al., 2018; Modrzejewski et al., 2020; Modrzejewski, 2020; Armengol-Estapé et al., 2020). This approach can also be adopted on the target-side, as presented here or in (Hoang et al., 2016a, 2018; Nguyen et al., 2018). However, these methods only add linguistic feature information to the input, without encouraging the system to model that information in any particular way.

Factored translation systems, under both statistical and neural machine translation, instead explore the addition of externally supplied linguistic features to the raw text at both input and output. These features include part-of-speech (POS) tags, word lemmatizations, morphological analysis, and semantic analysis (Koehn and Hoang, 2007; Garcia-Martinez et al., 2016, 2017; Tan et al., 2020). Factored translation models map feature-augmented input into feature-augmented output, however outputs include only an underlying lemma together with the predicted features. These systems also use a rule-based morphology toolkit in post-processing to generate the output surface forms from predicted output features, requiring knowledge of appropriate rule systems for the output language. An additional tagged architecture (Nădejde et al., 2017) predicted syntax-tagged surface forms, but did so by appending the tags to the surface form tokens directly, rather than predicting separate factors. In general, the focus of factored models has been to increase vocabulary coverage, for example of highly agglutitanative languages with rich morphologies, rather than our goal of disambiguating polysemous of polysyntactic words or otherwise handling named entities in a more nuanced way.

Finally, one previous work does consider a fully tagged (both source and target) factored neural model predicting tags with surface forms with independent layers in much the same way as presented here (Wagner, 2017). This work showed negative results for various syntactic tag types on IWSLT'14 shared task data (Cettolo et al., 2014), whereas this work presents NER and POS tags on film subtitles data.

3 Tagged seq2seq

We implemented two extensions to the standard seq2seq encoder-decoder architecture for neural machine translation to use token-level tags to improve translation results.[1] By combining token and tag embeddings in the input and simultaneously predicting tokens and tags in the output, the NMT

[1]Code at https://github.com/compwiztobe/tagged-seq2seq

system learned to translate tagged source sentences to tagged target sentences (Figure 1). We used a Transformer encoder and decoder for the base seq2seq model (Vaswani et al., 2017). Tags are added to the data as a preprocessing step.

3.1 Combined embedding

Learning an embedding for every possible token and tag combination would enormously increase the model's learnable parameter count. Furthermore, training data is likely to be sparse in its coverage of all possible pairs, but not in its coverage of the token and tag vocabularies separately. Therefore, we instead learn a separate embedding vector for each possible token and each possible tag, effectively concatenating these two vocabularies (rather than taking the product space). The embedding vectors for the token and tag at each position are then added to combine information from both channels into a single vector, so as not to increase the size of subsequent model layers and the capacity of the model, apart from the additional tag embedding vectors.

3.2 Simultaneous prediction

The decoder state d_i at each step is conditioned on the target prefix and the encoded source sentence (3).

$$d_i = \text{Decoder}(\text{prefix}, \text{src}) \tag{3}$$

This shared decoder state is used to predict both the next token and the next tag, with token and tag feature projections T and $\mathcal{T}$ (4 and 5).

$$P(\text{token } k \mid \text{prefix}; \text{src}) = \text{softmax}_k(T^\top d_i) \tag{4}$$

$$P(\text{tag } k \mid \text{prefix}; \text{src}) = \text{softmax}_k(\mathcal{T}^\top d_i) \tag{5}$$

We model these probabilities independently (6) for the same data sparsity and model size reasons as the embeddings, and we can compute each pair probability and loss accordingly (7).

$$\begin{aligned} &P(\text{token}, \text{tag} \mid \text{prefix}; \text{src}) \\ &\quad = P(\text{token} \mid \text{pre.}; \text{src}) \cdot P(\text{tag} \mid \text{pre.}; \text{src}) \end{aligned} \tag{6}$$

$$\begin{aligned} \mathcal{L} = &- \log P(\text{token} \mid \text{prefix}; \text{src}) \\ &- \log P(\text{tag} \mid \text{prefix}; \text{src}) \end{aligned} \tag{7}$$

This combined loss encourages the shared decoder state d_i to model the correct tag identity so that it can be used by the token prediction layer to improve translation.

4 Data Preparation

4.1 Subtitles corpus

Our experiments focused on film subtitles in German and English. The Opus project provided a parallel German to English subtitles corpus from OpenSubtitles (Tiedemann, 2012; Aulamo et al., 2020). This data was cleaned with some rudimentary sentence length filtering, and randomly divided into a 3 million sentence-pair training split (about 49 million tokens), along with 100,000 pair validation and test splits (about 1.6 million tokens each).

4.2 Tagging "off the shelf"

Flair NLP tools systems have achieved state-of-the-art results on the sequence labeling tasks such as the CoNLL'03 NER dataset and universal part-of-speech tagging from Universal Dependency treebanks (Akbik et al., 2018; Tjong Kim Sang and De Meulder, 2003; Nivre et al., 2020). We used the publicly available pre-trained multilingual NER and universal POS taggers.[2] NER tags followed the `BIOES` system with four entity classes: `PER`, person; `LOC`, location; `ORG`, organization; and `MISC`, miscellaneous. Four classes with four span markers, plus the null span marker `O`, gave the same 17-tag vocabulary for NER on both German and English. Meanwhile, POS tags came from the same 17-tag universal POS tag set for both languages.

Around 3% of words in the OpenSubtitles corpus were tagged as named entities (non `O`). We further divided the test split based on whether any named entities were found in either the source or the target sentence. Out of 100,000 test pairs, 79,201 had no named entities, and 20,799 had some.

4.3 Tokenization

Word tokenization, as used by the tagging systems, is most straightforward for maintaining one-to-one alignments between tokens and their assigned tags. For word tokenization experiments, vocabularies of size 35,012 for German and 17,196 for English were selected, resulting in an unknown word replacement rate of 3%.

This unknown word replacement was considerably higher on rare word categories, for example named entities saw a 25 – 30% rate of unknown words outside the selected word vocabulary. To alleviate this it is also possible to consider subword

[2]Models at `https://huggingface.co/flair/{ner,upos}-multi`

Table 1: BLEU scores on word-tokenized sentences with or without named entities, for models with or without NER tags.

	BLEU (%)		
NER tags	no NEs	some NEs	all
−src, −tgt[3]	34.70	32.43	34.15
+src, −tgt[4]	*34.89*	*32.14*	*34.22*
−src, +tgt[5]	*35.69*	*35.03*	*35.53*
+src, +tgt	**35.84**	**35.50**	**35.76**
improvement	↑ **1.14**	↑ **3.07**	↑ **1.61**

tokenization, so additional experiments were conducted with a shared SentencePiece (Kudo, 2018) vocabulary of 32,000 subwords, built from the training split and used to tokenize both languages.

After subword tokenization, the `BIOES` structure of named entity spans was propagated across subword tokens in the natural way to maintain spans. For POS tags, subwords received the same tag as their parent word.

5 Experiments

We used a Transformer encoder and decoder (Vaswani et al., 2017) for the base seq2seq system, each with 6 layers and 8 attention heads, and layer and embedding dimensions 512. Training was done for 40 epochs at half precision with the optimizer known as Adam (Kingma and Ba, 2015) with $\beta = (0.9, 0.98)$ and an inverse square root learning schedule with maximum learning rate 5×10^{-4} after 500 updates and decay 1×10^{-4}. Parameter updates occurred after every 8,192 token-tag pairs at most (rounding off to complete sentences), with 30% dropout and label smoothing of 0.1 on the training loss.

At inference time, a beam of 5 candidates was maintained, and the models were evaluated with their BLEU score on the token sequence only (tagging accuracy was not evaluated due to the difficulty of establishing alignment).

6 Results

BLEU scores from untagged and tagged translation experiments show an improvement from the use of NER tags (Table 1). Adding NER tags, the BLEU score on sentences containing some named entities improved by a larger margin, 3.07 points, presumably due to the tags' assistance with translating those named entities. We also note an improvement in the BLEU score on sentences containing no named entities, which increased by 1.14 points. This suggests that given `O` tag information the model can also treat common words with confidence that they are not named entities and should not be translated as such. These improvements averaged out to a net gain of 1.61 BLEU points on the entire test split.

Table 2: BLEU scores for word models with POS tags.

POS tags	BLEU (%)
−src, −tgt	34.15
+src, −tgt	***34.21***
−src, +tgt	*33.70*
+src, +tgt	33.77
improvement	↓ 0.38

We also evaluated a model trained with POS tags, but found a decrease in BLEU score (Table 2). Translation scores with POS tags decreased by 0.38 BLEU points. There are two ways to understand this in comparison with NER tags. First, POS tags carry a significant amount of information about the sentence, not only helping to disambiguate between different word senses by part-of-speech, but also assisting the model with encoding the sentence's syntactic structure. Compared to NER tags, this amount of structural information might be difficult to model with the same decoder architecture used for token prediction. Second, POS tags tend to carry the same amount of information for each tag at each position, compared to NER tags only conveying most of their information at the named entity spans which are few and far between. This also lends itself to the idea that POS tags have a higher information content that is less easily modeled by the decoder, leading to worse results than NER tagging.

6.1 Enhanced baselines and ablation study

For both NER and POS tagged results, the baseline was the same Transformer architecture trained only on untagged data (without adding tag embeddings or predicting tags from the decoder). Adding in only source-side tag embeddings could be considered an enhanced baseline, since this kind of

[3] baseline
[4] enhanced baseline / ablation study
[5] ablation study

Table 3: BLEU scores on subword-tokenized sentences with or without named entities, for models with or without NER tags.

	BLEU (%)		
NER tags	no NEs	some NEs	all
−src, −tgt	35.77	36.51	35.96
+src, −tgt	*35.83*	*36.75*	*36.06*
−src, +tgt	*35.88*	*36.82*	*36.12*
+src, +tgt	**35.94**	**36.92**	**36.19**
improvement	↑ **0.17**	↑ **0.41**	↑ **0.22**

Table 4: BLEU scores for subword models with or without POS tags.

POS tags	BLEU (%)
−src, −tgt	35.96
+src, −tgt	***36.20***
−src, +tgt	*35.69*
+src, +tgt	35.74
improvement	↓ 0.22

feature augmentation has already been studied in depth (Sennrich and Haddow, 2016; Hoang et al., 2016b). Our results show that this source-only tagging does not provide significant benefits compared to training on untagged data (Table 1), although for POS tagging this remains the best result.

On the other hand, adding in target-side tags while also predicting them from the decoder, without adding in source-side tag embeddings could be considered an ablation test to isolate the effects of our main contribution: target-side tag decoding. Our results show that this target tagging provides the same benefit as the fully tagged training regime, demonstrating that it is the simultaneous tag decoding that accounts for the entire effect observed. For NER tagging this was an improvement in BLEU scores, but for POS tagging scores decreased when adding target tagging.

Whereas source-side tag information is added into the embeddings without any modification to the training objective, target-side tag predictions are a part of the modified training loss, so that it is the target-side tag prediction that pushes the model to incorporate accurate knowledge of the tags into its learning representations. That NER tag modeling improved results while POS tag modeling did not is consistent with our earlier observation that POS tag modeling seems to be more difficult than NER tag modeling, and is not done effectively by the current architecture.

6.2 Subword tokenization experiments

Experiments with subword tokenized data showed similar effects, but of a significantly reduced size. Adding NER tags improved the results, adding 0.22 points to the BLEU score, with the improvement again coming largely from the target side tagging, and again showing a larger improvement on sentences with named entities than on those without (Table 3). Adding POS tags hurt results, decreasing the score by 0.22, and again we see that source-only tagging is best case for POS tagging (Table 4). However, the reduced magnitude of these deltas to the range of 0.1 – 0.4 BLEU points suggests these are not significant changes to the translation performance, in the subword tokenization case.

It would appear that subword tokenization interferes with the benefits of tagging the data. Since tags are aligned one-to-one with the input words, subword tokenization destroys this alignment, and copying tags across a word's constituent subwords may interfere with the model's ability to make sense the of tag information. In particular for named entities, rare words are likely to tokenized into a larger number of subword tokens, exacerbating this effect. The set of embeddings for the subwords in a word may not be as useful to the model for translating a named entity or other rare category as the single embedding learned specifically for the full word in a word tokenization setting, and further these subword embeddings may be affected by other contexts unrelated to the larger word. Specifically for the named entity case, subword tokenization algorithms might prioritize the atomicity of certain rare words tagged as named entities in order to counteract this.

6.3 Token prediction and tagging loss

Due to the conditional independence assumption, the cross-entropy loss (7) conveniently decomposes into separate terms for tokens and tags (8), allowing us to measure the relative information content of each channel (Table 5).

$$\begin{aligned}\mathcal{L} &= -\log P(\text{token} \mid \text{prefix; src}) \\ &\quad -\log P(\text{tag} \mid \text{prefix; src}) \\ &= \mathcal{L}_{\text{token}} + \mathcal{L}_{\text{tag}}\end{aligned} \tag{8}$$

Table 5: Token prediction and tagging loss.

		↓ cross entropy (bits)		
		$\mathcal{L}_{\text{token}}$	$\mathcal{L}_{\text{tag}}$	$\mathcal{L}$
no tags	−src, −tgt	2.000	—	2.000
NER	+src, −tgt	2.006	—	2.006
	−src, +tgt	2.001	0.183	2.184
	+src, +tgt	**1.985**	0.183	2.168
POS	+src, −tgt	2.007	—	2.007
	−src, +tgt	1.995	*0.697*	2.692
	+src, +tgt	**1.972**	*0.695*	2.673

While adding tag information naturally increases the overall cross-entropy, as there are more possibilities to account for and to be predicted, restricting our attention only to the token loss shows that the token-level cross-entropy is consistently reduced from 2.000 (base-2) to 1.985 with NER tags or 1.972 for POS tags. This shows how both tag types can add disambiguating information to the token prediction process, with POS tags naturally add more of such information, since they carry syntactic information.

Looking only at tag-level cross-entropy, it's interesting to notice that the POS tagging loss is significantly higher than the NER tagging loss. While this could be simply because the lower-bound inherent entropy is higher (POS tags naturally contain more information, being more uniformly distributed than NER tags), this could also be consistent with the idea that POS tag modeling is more difficult, explaining the decreased translation scores observed with POS tag prediction.

7 Model Limitations

It should not go unnoticed that the typical inference algorithms for sequence labeling, particularly the BiLSTM-CRF inference employed by most NER systems, are incompatible with the autoregressive sequence decoding algorithms (greedy decoding and beam search) used for inference by seq2seq models. That the beam decoding algorithm (and autoregressive likelihood model) used here for tags was unable to account for (be conditioned on) the as-yet uncomputed right context was cause for much apprehension before experimental results became available. These positive results notwithstanding, future work could explore how to better incorporate the full tagging context in tag decoding, perhaps, for example, by predicting the sequence more wholistically with non-autoregressive decoding (Gu et al., 2018).

We also imagine that the design of the underlying seq2seq architecture may lend itself to certain types of sequence labeling. For example, the bidirectional context modeled by a BiLSTM-based translation model may be more suitable for certain types of sequence labeling tasks than the Transformer's attentional activations. Because our contributions are agnostic to the type of sequence labeling (NER or part-of-speech tagging or any other kind) as well as to the design of the encoder and decoder, future experiments should also explore these possibilities.

8 Conclusion

We implemented extensions to existing neural machine translation models that allow the use of off-the-shelf token-level tagging systems to improve translation accuracy. Translation inputs and training outputs were tagged with pre-trained sequence labeling systems. A standard encoder-decoder architecture was extended to include tag embeddings and tag prediction at each token position. At model input, token and tag embedding vectors were added to produce a combined embedding. At model output, the final decoder layer used separate softmax layers to predict tokens and tags. During training, a combined loss function encouraged the model to learn token and tag information jointly.

This tag assisted translation system was tested against baseline token-only systems on a German to English film subtitle corpus with both word and subword tokenization. Subword tokenization reduced the size of the effect, suggesting the need for specialized subword tokenization to prioritize the integrity of important word categories. However, on word tokenized data, the 1.61 point increase in BLEU score using named entity tags demonstrates that the proposed architecture is useful for improving translation outputs with automatic named entity recognition, while the 0.38 point decrease using part-of-speech tags indicates more difficulty in utilizing that tag information. Further examination of the cross-entropy showed that adding tags reduced the token cross-entropy thereby improving token modeling. Future experiments can explore the use of other types of tag data as well as other decoding paradigms.

Acknowledgments

Many thanks go to my colleagues Jeesoo Bang, Jaehun Shin, and Baikjin Jung in the Knowledge and Language Engineering Lab (POSTECH) for their many hours generously spent discussing these research topics. These results would not have been possible without their support.

This work was carried out as part of the HPC Support Project supported by the Ministry of Science and ICT (MSIT) and the National IT Industry Promotion Agency (NIPA), and was funded by the Institute of Information & Communications Technology Planning & Evaluation (IITP) supported by the Korean government (MSIT): Grant No. 2019-0-01906, Graduate School of Artificial Intelligence (POSTECH).

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1638–1649, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Jordi Armengol-Estapé, Marta R Costa-jussà, and Carlos Escolano. 2020. Enriching the transformer with linguistic factors for low-resource machine translation. *arXiv preprint arXiv:2004.08053*.

Mikko Aulamo, Umut Sulubacak, Sami Virpioja, and Jörg Tiedemann. 2020. OpusTools and parallel corpus diagnostics. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3782–3789, Marseille, France. European Language Resources Association.

Bogdan Babych and Anthony Hartley. 2003. Improving machine translation quality with automatic named entity recognition. In *Proceedings of the 7th International EAMT workshop on MT and other language technology tools, Improving MT through other language technology tools, Resource and tools for building MT at EACL 2003*.

M. Cettolo, J. Niehues, S. Stüker, L Bentivogli, and Marcello Federico. 2014. Report on the 11th IWSLT evaluation campaign. In *Proceedings of the 11th International Workshop on Spoken Language Translation*, pages 2–16.

André Ferreira Cruz, Gil Rocha, and Henrique Lopes Cardoso. 2020. Coreference resolution: Toward end-to-end and cross-lingual systems. *Information*, 11:74.

Mercedes Garcia-Martinez, Loïc Barrault, and Fethi Bougares. 2016. Factored Neural Machine Translation Architectures. In *International Workshop on Spoken Language Translation (IWSLT'16)*, Seattle, United States.

Mercedes Garcia-Martinez, Loïc Barrault, and Fethi Bougares. 2017. Neural Machine Translation by Generating Multiple Linguistic Factors. In *5th International Conference Statistical Language and Speech Processing SLSP 2017*, Statistical Language and Speech Processing 5th International Conference, SLSP 2017, Le Mans, France, October 23–25, 2017, Proceedings, Le Mans, France. 11 pages, 3 figues, SLSP conference.

Jiatao Gu, James Bradbury, Caiming Xiong, Victor O. K. Li, and Richard Socher. 2018. Non-autoregressive neural machine translation. In *6th International Conference on Learning Representations, ICLR 2018, Conference Track Proceedings*, Vancouver, BC, Canada.

Cong Duy Vu Hoang, Trevor Cohn, and Gholamreza Haffari. 2016a. Incorporating side information into recurrent neural network language models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1250–1255, San Diego, California. Association for Computational Linguistics.

Cong Duy Vu Hoang, Gholamreza Haffari, and Trevor Cohn. 2016b. Improving neural translation models with linguistic factors. In *Proceedings of the Australasian Language Technology Association Workshop 2016*, pages 7–14, Melbourne, Australia.

Cong Duy Vu Hoang, Gholamreza Haffari, and Trevor Cohn. 2018. Improved neural machine translation using side information. In *Proceedings of the Australasian Language Technology Association Workshop 2018*, pages 6–16, Dunedin, New Zealand.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, Conference Track Proceedings*, San Diego, CA, USA.

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. Association for Computational Linguistics.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75, Melbourne, Australia. Association for Computational Linguistics.

J. Li, A. Sun, J. Han, and C. Li. 2020. A survey on deep learning for named entity recognition. In *IEEE*

Transactions on Knowledge and Data Engineering, Los Alamitos, CA, USA. IEEE Computer Society.

Maciej Modrzejewski. 2020. *Improvement of the Translation of Named Entities in Neural Machine Translation*. Ph.D. thesis, Karlsruhe Institute of Technology Department of Informatics Institute for Anthropomatics and Robotics.

Maciej Modrzejewski, Miriam Exel, Bianka Buschbeck, Thanh-Le Ha, and Alexander Waibel. 2020. Incorporating external annotation to improve named entity translation in NMT. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 45–51, Lisboa, Portugal. European Association for Machine Translation.

Maria Nădejde, 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, Copenhagen, Denmark. Association for Computational Linguistics.

Quang-Phuoc Nguyen, Joon-Choul Shin, and Cheol-Young Ock. 2018. An evaluation of translation quality by homograph disambiguation in korean-x neural machine translation systems. In *Annual Conference on Human and Language Technology*, pages 504–509. Human and Language Technology.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France. European Language Resources Association.

Rico Sennrich and Barry Haddow. 2016. Linguistic input features improve neural machine translation. In *Proceedings of the First Conference on Machine Translation: Volume 1, Research Papers*, pages 83–91, Berlin, Germany. Association for Computational Linguistics.

Samson Tan, Shafiq Joty, Lav Varshney, and Min-Yen Kan. 2020. Mind your inflections! Improving NLP for non-standard Englishes with Base-Inflection Encoding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5647–5663, Online. Association for Computational Linguistics.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eight International Conference on Language Resources and Evaluation*, Istanbul, Turkey. European Language Resources Association.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, page 142–147, USA. Association for Computational Linguistics.

Arata Ugawa, Akihiro Tamura, Takashi Ninomiya, Hiroya Takamura, and Manabu Okumura. 2018. Neural machine translation incorporating named entity. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3240–3250, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pages 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Martin Wagner. 2017. *Target Factors for Neural Machine Translation*. Ph.D. thesis, Karlsruhe Institute of Technology Department of Informatics Institute for Anthropomatics and Robotics.

Zihan Wang, Jingbo Shang, Liyuan Liu, Lihao Lu, Jiacheng Liu, and Jiawei Han. 2019. CrossWeigh: Training named entity tagger from imperfect annotations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5154–5163, Hong Kong, China. Association for Computational Linguistics.