Bring More Attention to Syntactic Symmetry for Automatic Postediting of High-Quality Machine Translations

Baikjin Jung[◇] **Myungji Lee**[♡] **Jong-Hyeok Lee**[◇♡] **Yunsu Kim**[◇♡]

[◇]Department of Computer Science and Engineering
[♡]Graduate School of Artificial Intelligence
Pohang University of Science and Technology, Republic of Korea
{bjjung, mjlee7, jhlee, yunsu.kim}@postech.ac.kr

Abstract

Automatic postediting (APE) is an automated process to refine a given machine translation (MT). Recent findings present that existing APE systems are not good at handling high-quality MTs even for a language pair with abundant data resources, English–German: the better the given MT is, the harder it is to decide what parts to edit and how to fix these errors. One possible solution to this problem is to instill deeper knowledge about the target language into the model. Thus, we propose a linguistically motivated method of regularization that is expected to enhance APE models' understanding of the target language: a loss function that encourages symmetric self-attention on the given MT. Our analysis of experimental results demonstrates that the proposed method helps improving the state-of-the-art architecture's APE quality for high-quality MTs.

1 Introduction

Automatic postediting (APE) is an automated process to transform a given machine translation (MT) into a higher-quality text (Knight and Chander, 1994). Since 2015, Conference on Machine Translation (WMT) has been hosting an annual shared task for APE, and most of the recently developed APE systems are within the common framework of representation learning using artificial neural networks to learn postediting patterns from the training data (Chatterjee et al., 2018, 2019, 2020; Akhbardeh et al., 2021).

Since 2018, all participants in the shared task have used Transformer-based models (Vaswani et al., 2017), but recent findings of the shared task (Chatterjee et al., 2018, 2019, 2020; Akhbardeh et al., 2021) cast doubt on whether Transformer-based APE models learn good generalizations because such models' APE quality appears to be significantly affected by external factors such as the source–target language pair, the qualitative characteristics of the provided data, and the quality of the given MT.

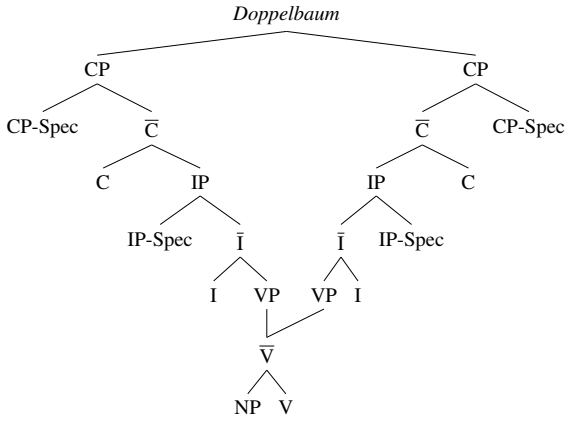

Figure 1: A depiction of *Doppelbaum* (§2).

Especially, the good quality of the given MTs has brought great difficulty in performing APE on the WMT 2019 test data set: the better the given MT is, the harder it is to decide what parts to edit and how to correct these errors (Chatterjee et al., 2018, 2019). The thing to notice is that this outcome is not a question of data scarcity because the language pair of this test data set, English–German, is a language pair provided with abundant training, validation, and test data. Also, it is not a question of data heterogeneity, either: the domain of this test data set, IT, shows a high degree of lexical repetition, which indicates that data sets in this domain use the same small set of lexical items (Chatterjee et al., 2018, 2019; Akhbardeh et al., 2021). Thus, it would be a question of modeling, and one possible solution is to implant deeper knowledge about the target language into the model.

To this end, we propose a new method of regularization that is expected to enhance Transformer-based APE models' understanding of German translations. Specifically, the proposed method is based on *Feldermodell* (§2), an established linguistic model, which implies the need for proper treatment

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 2: Short Papers, pages 1433–1441

of the underlying symmetry of German sentence structures. To instill the idea of syntactic symmetry into Transformer-based APE models, we introduce a loss function that encourages symmetric self-attention on the given MT. Based on experimental results, we conduct a careful analysis and conclude that the proposed method has a positive effect on improving the state-of-the-art architecture's APE quality for high-quality MTs.

2 Linguistic Theory

In German linguistics, *das topologische Satzmodell* ('the topological sentence model') or *das Feldermodell* ('the field model') (Reis, 1980; Wöllstein, 2018; Höhle, 2019) describes how constituents of a sentence are closely related even if they are far apart from each other. Usually, *Feldermodell* divides a clause into *das Vorfeld* ('the prefield'; VF), *die linke Satzklammer* ('the left bracket'; LSK), *das Mittelfeld* ('the middlefield'; MF), *die rechte Satzklammer* ('the right bracket'; RSK), and *das Nachfeld* ('the postfield'; NF).

(1) [Heute $_{\text{VF}}$] [habe $_{\text{LSK}}$] [ich $_{\text{MF}}$] [gesehen $_{\text{RSK}}$] [zufällig $_{\text{NF}}$],

(2) [[dass $_{\text{LSK}}$] [du eine Tasse Kaffee $_{\text{MF}}$] [getrunken hast $_{\text{RSK}}$] $_{\text{NF}}$].

These parts are all interrelated; LSK and RSK are a typical example: while the former holds a finite verb or a complementizer, the latter holds a past participle, an infinitive, and a particle. In (1), VF holds "*Heute*" ('today'); LSK holds "*habe*" ('have'); MF holds "*ich*" ('I'); RSK holds "*gesehen*" ('seen'); and NF holds "*zufällig*" ('by chance'). (2) is an additional NF of (1) and includes its own LSK holding "*dass*" ('that'); MF holding "*du eine Tasse Kaffee*" ('you a cup of coffee'); and RSK holding "*getrunken hast*" ('drank').

For such analyses, special tree structures such as *Doppelbaum* (Wöllstein, 2018) ('double tree') can be used, which is a bimodal tree (Fig. 1), where two CP, $\overline{\text{C}}$, IP, $\overline{\text{I}}$, and VP subtrees are '**symmetric**' with respect to $\overline{\text{V}}$. We assume that this structural symmetry is parameterized from the perspective, not only of generative linguistics (Wöllstein, 2018; Höhle, 2019), but also of a parametric model $\mathcal{P} = \{P_\theta \mid \theta \in \Theta\}$, where P_θ and Θ are a probability distribution and the parameter space, respectively.

Especially, if we look at APE in terms of sequence-to-sequence learning (Sutskever et al., 2014), the probability distribution of the output sequence $(y_1, \cdots, y_{L_y})$ is obtained in the following manner:

$$P_\theta(y_1, \cdots, y_{L_y} \mid x_1, \cdots, x_{L_x}, z_1, \cdots, z_{L_z}) = \prod_{t=1}^{L_y} P_\theta(y_t \mid u, v, y_1, \cdots, y_{t-1}),$$

where u and v are the representations of a source text $(x_1, \cdots, x_{L_x})$ and its MT $(z_1, \cdots, z_{L_z})$, respectively. In this process, we presume that the syntactic symmetry of the target language affects the resulting distribution P_θ; in other words, this syntactic symmetry would be an inductive bias (Mitchell, 1980) that should be handled properly.

3 Methodology

We implement a multi-encoder Transformer model consisting of the "Joint-Final" encoder and the "Parallel" decoder, which is a state-of-the-art architecture for APE (Shin et al., 2021), and conduct a controlled experiment without concern for usage of performance-centered tuning techniques. Specifically, the Joint-Final encoder consists of a source-text encoder and an MT encoder, which process the given source text and MT, respectively. Based on this baseline architecture, we propose a method to encourage the MT encoder to perform symmetric self-attention by minimizing the skewness of each self-attention layer's categorical distribution p_{self}.

The used measure of skewness is

$$(\ddot{\mu}_3)_i = \left(\sum_{j=1}^{\lfloor \frac{L_z}{2} \rfloor} p_{\text{self}}[i,j] - \sum_{j=\lfloor \frac{L_z}{2} \rfloor + 1}^{L_z} p_{\text{self}}[i,j] \right)^2,$$

for each token z_i in the given MT $(z_1, \cdots, z_{L_z})$.

Accordingly, the basic cross-entropy loss $\mathcal{L}_{\text{CE}}$ is regularized by $(\ddot{\mu}_3)_i$, resulting in a new loss function

$$\mathcal{L}_{\text{DOPPELBAUM}} = \mathcal{L}_{\text{CE}} + \mathbb{E}[\alpha]\mathbb{E}[\ddot{\mu}_3] + (1 - \alpha),$$

where

$$\mathbb{E}[\alpha] = \frac{\sum_{b=1}^{B} \sum_{i=1}^{L_z} \alpha_{b,i}}{B \times L_z}$$

is the expected value of coefficients

$$\alpha_{b,i} = \sigma(W^{\mathrm{T}} v_{b,i} + \beta)$$

in the given minibatch, and

$$\mathbb{E}[\ddot{\mu}_3] = \frac{\sum_{b=1}^{B} \sum_{n=1}^{N} \sum_{h=1}^{H} \sum_{i=1}^{L_z} (\ddot{\mu}_3)_{b,n,h,i}}{B \times N \times H \times L_z}$$

is the expected value of $(\ddot{\mu}_3)_{b,n,h,i}$. In addition, $(1-\alpha)$ is an initial inducement to utilizing $\ddot{\mu}_3$. In the equations above, σ is the sigmoid function, v is the output of the final layer of the MT encoder, $W \in \mathbb{R}^{d_{\text{model}}}$ and $\beta \in \mathbb{R}$ are learned parameters, B is the number of data examples, N is the number of layers, and H is the number of heads.

4 Experiment

In the conducted experiment, all hyperparameters are the same as those of Shin et al. (2021) except the learning rate (Appendix A); we basically reproduce their experimental design.

Data Sets		Sizes
Training	eSCAPE-NMT	5,065,187
	WMT 2019	13,442
Validation	WMT 2019	1,000
Test	WMT 2019	1,023

Table 1: APE data sets used in the experiment. eSCAPE-NMT is a cleansed subset of eSCAPE's (Negri et al., 2018) English–German-NMT set. The cleansing procedure is a reproduction of Shin et al. (2021). The WMT 2019 data sets (Chatterjee et al., 2019) were released for WMT 2018 but used also at WMT 2019.

Both the baseline model and the proposed model are trained by using the training data sets and the validation data set listed in Table 1; we first train the models by using eSCAPE-NMT mixed with the WMT 2019 training data in the ratio of 27 : 1, and then tune them by using the WMT 2019 training data solely.

5 Results and Analysis

The result of automatic evaluation (Table 2) indicates that the proposed model improves on the baseline model in terms of BLEU (75.47) but does not in terms of TER (16.54), which is unusual. Although those measures have a strong correlation overall (Fig. 2), the proposed model has more outliers, δBLEU (the value obtained by subtracting a given MT's BLEU from the postedited result's BLEU) of which is over 20, compared to the baseline model; they must be the ones that bring the improvement in BLEU.

Thus, we present an additional evaluation result to further investigate this mismatch between TER improvements and BLEU improvements: a relative frequency distribution of successes and failures in APE with regard to the TER difference between a given MT and each model's output (Table 3). Then, the mentioned outliers correspond to PERF, which is the set of the cases where an APE system succeeds in perfectly correcting the given MT with one or more errors, considering that the proposed model's PERF has a $\mu_{\delta\text{BLEU}}$ (the average of sentence-level BLEU improvements) of 27.21. We see that the proposed model has substantially more PERF cases (5.87%) than the baseline model (4.30%) and that because most of those 'new' (1.57*pp*) cases are results of nontrivial postediting (Table 4), this increase in the proportion of perfect postediting is valid evidence of the proposed method's effect on enhancing the baseline model's APE quality for high-quality MTs.

Systems	WMT 2019	
	TER$^{\downarrow}$ (σ)	BLEU$^{\uparrow}$ (σ)
Given MT	16.84 (19.52)	74.73 (25.89)
Baseline	16.60† (19.51)	75.11† (26.21)
DOPPELBAUM	**16.54**† (19.48)	**75.47**$^{\dagger *}$ (26.16)

Table 2: The results of automatic evaluation on the WMT 2019 test data set. Baseline is the above-mentioned baseline model (§3), and DOPPELBAUM is the proposed model. Beside TER (Snover et al., 2006) and BLEU (Papineni et al., 2002), their sentence-level standard deviations (σ) are presented. In each column, the figure implying the best performance is in **bold**. The dagger symbols denote the proposed model's quality improvement on the given MTs is statistically significant ($p \leq 0.05$). The asterisks denote the proposed model's performance improvement on the baseline model is statistically significant ($p \leq 0.05$).

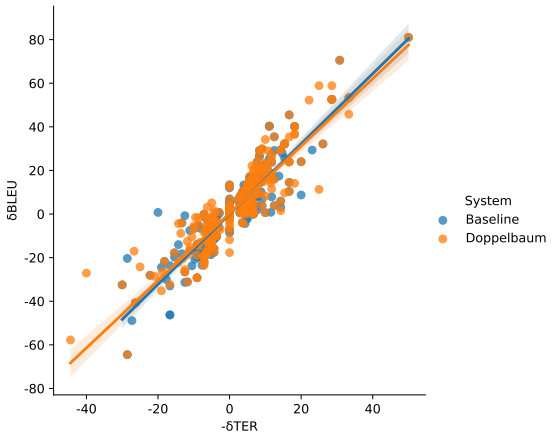

Figure 2: The relationship between models' sentence-level TER improvements ($-\delta$TER; positive values denote decrease in TER) and sentence-level BLEU improvements (δBLEU; positive values denote increase in BLEU) on those of the given MTs in the test data set.

Systems		Modified					Intact		F1
		Ruin	Degr	Even	Impr	Perf	Acce	Negl	
Baseline	%	1.86	**6.65**	5.67	**7.14**	4.30	23.36	51.03	
	$\mu_{\delta\text{BLEU}}$	−24.48	−13.51	0.50	9.22	27.23	0.00	0.00	22.8
	$\sigma_{\delta\text{BLEU}}$	15.48	9.42	3.38	8.43	16.39	0.00	0.00	
Doppelbaum	%	**1.56**	7.33	5.77	**7.14**	**5.87**	**23.66**	**48.68**	
	$\mu_{\delta\text{BLEU}}$	−26.12	−11.72	−0.42	10.04	27.21	0.00	0.00	**25.4**
	$\sigma_{\delta\text{BLEU}}$	16.09	9.16	3.82	8.69	16.37	0.00	0.00	

Table 3: A relative frequency distribution containing the frequencies of the following groups (we compare the TER of the given MT and that of the postedited result.): the cases where an APE system injects errors to an already perfect MT (Ruin); both the given MT and the APE result are not perfect, but the former is better in terms of TER (Degr); both are not perfect and have the same TER although they are different from each other (Even); both are not perfect, but the latter is better (Impr); the given MT is not perfect whereas the APE result is (Perf); both are perfect (Acce); and lastly, even though the MT is not perfect, the APE system does not change anything (Negl). The calculation of the F1 score is based on two criteria: whether the given MT is perfect or not (for recall) and whether the APE system edits the given MT or not (for precision). % is the proportion of the cases belonging to each category, $\mu_{\delta\text{BLEU}}$ is the average of sentence-level BLEU improvements, and $\sigma_{\delta\text{BLEU}}$ is their standard deviation.

Types of APE			Numbers
Perf	Linguistic	Nouns	5
		Expressions	5
		Agreement	3
		Prepositions	2
	Other	Punctuation	5
		URLs	2
		Noise Removal	2
		Total	24
Acce	Linguistic	Nouns	3
		Expressions	2
		Adjectives	1
	Other	Punctuation	2
		Total	8

Table 4: Manual categorization of the cases where only the proposed model produces a perfect translation. For more information on the definitions of Perf and Acce, refer to Table 3. 'Linguistic' and 'Other' cases are results of nontrivial postediting and trivial postediting, respectively. 'Expressions' means using appropriate determiners, verb phrases, shortened forms of the definite article, etc.. 'Noise Removal' means filtering out meaningless tokens from the given MT. This categorization was double-checked by a native German speaker.

In addition, in an actual example where only the proposed model corrects the given MT perfectly (Table 5), we observe that the proposed model successfully captures the close relation between the verb "*enthält*" ('contains') and its object so that the correct form "*Variablen*" ('variables') is used. Considering that the adverb phrase "*zum Beispiel*" ('for example') in the given MT makes some distance between the verb and its object, it appears that the proposed model integrates information from a wider range of constituents than the baseline model; hence the conclusion that the proposed method instills *Feldermodell*'s idea of syntactic symmetry into Transformer-based APE models and enhances their understanding of German translations.

Another example (Table 6) suggests that the increase in the proportion of Acce (0.3*pp*), which is the set of the cases where an APE system adopts the given, already perfect MT, should be cautiously interpreted. Although professional translators tend to perform "only the necessary and sufficient corrections" (Bojar et al., 2015), the validity of test data created by professional translators, including the WMT 2019 test data set, can also be disputable because other native speakers might argue that they can perform better postediting. For example, some people may consider hyphenated compound "*Zoom-Werkzeug*" ('Zoom tool') more natural than closed compound "*Zoomwerkzeug*" (Table 6).

However, considering the big differences in the proportion of Negl (2.35*pp*), which is the set of the cases where an APE system neglects to postedit the given MT, and the F1 score (Table 3), it appears that such a risk need not be considered in this analysis. Moreover, the proposed model has fewer Ruin cases (1.56%), where it injects errors to the given, already perfect MT, than the baseline model (1.86%). Although the proposed model has more Degr cases (7.33%), where it degrades the given MT, than the baseline

Case 1: Perf		TER↓	Bleu↑
Source Text	For example , the following function contains variables that are defined in various block scopes .		
Given MT	Die folgende Funktion enthält zum Beispiel Variable , die in verschiedenen Codebereichen definiert sind .	6.67	80.03
Baseline	Die folgende Funktion enthält zum Beispiel Variable , die in verschiedenen Codebereichen definiert sind .	6.67	80.03
Doppelbaum	Die folgende Funktion enthält zum Beispiel Variablen , die in verschiedenen Codebereichen definiert sind .	0.00	100.00
Manual Postediting	Die folgende Funktion enthält zum Beispiel Variablen , die in verschiedenen Codebereichen definiert sind .		

Table 5: A case where only the proposed model corrects the given MT perfectly. Considering the manually postedited result, wrong words in the given MT, the APE result of the baseline model, and that of the proposed model are highlighted in pink while correct words are highlighted in green. All the texts are tokenized or detokenized using Moses (Koehn et al., 2007).

Case 2: Acce		TER↓	Bleu↑
Source Text	Double-click the Zoom tool .		
Given MT	Doppelklicken Sie auf das Zoomwerkzeug .	0.00	100.00
Baseline	Doppelklicken Sie auf das Zoom-Werkzeug .	16.67	53.73
Doppelbaum	Doppelklicken Sie auf das Zoomwerkzeug .	0.00	100.00
Manual Postediting	Doppelklicken Sie auf das Zoomwerkzeug .		

Table 6: A case where only the proposed model adopts the given, already perfect MT. Details are the same as in Table 5.

(6.65%), the proposed model's quality degradation $\mu_{\delta\text{BLEU}} = -11.72$ is less severe than that of the baseline ($\mu_{\delta\text{BLEU}} = -13.51$). Therefore, we conclude that the proposed method results in small but certain improvements.

6 Conclusion

To improve the APE quality for high-quality MTs, we propose a linguistically motivated method of regularization that enhances Transformer-based APE models' understanding of the target language: a loss function that encourages APE models to perform symmetric self-attention on a given MT. Experimental results suggest that the proposed method helps improving the state-of-the-art architecture's APE quality for high-quality MTs; we also present a relative frequency distribution of successes and failures in APE and see increases in the proportion of perfect postediting and the F1 score. This evaluation method could be useful for assessing the APE quality for high-quality MTs in general. Actual cases support that the proposed method successfully instills the idea of syntactic symmetry into APE models. Future research should consider different language pairs and different sets of hyperparameters.

7 Acknowledgements

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (Ministry of Science and ICT) (No. 2019-0-01906, Artificial Intelligence Graduate School Program (POSTECH)). We thank Richard Albrecht for assistance in the manual categorization of cases.

8 Limitations

First, neither *Feldermodell* (Reis, 1980; Wöllstein, 2018; Höhle, 2019) nor *Doppelbaum* (Wöllstein, 2018) has obtained complete concurrence among linguists. Also, we limit our scope to the English–German language pair and the IT domain using the WMT 2019 training, validation, and test data sets. A broader scope would not provide confidence in the validity of conducted experiments because there are hardly any standard setups for experimental research (Chatterjee et al., 2018, 2019; Akhbardeh et al., 2021).

In addition, the conducted experiment should take into consideration the effect of randomness that is attended in the process of training artificial neural networks; different techniques, different hyperparameters, and multiple runs of optimizers (Clark et al., 2011) may present different results. However, as previous studies (Chatterjee et al., 2018, 2019, 2020; Akhbardeh et al., 2021), including the study on the baseline model (Shin et al., 2021), do not consider the effect of randomness, we also do not investigate the effect of randomness further, considering that training multiple models (Appendix A) to obtain good estimators (TER and BLEU) will cost a lot.

References

Farhad Akhbardeh, Arkady Arkhangorodsky, Magdalena Biesialska, Ondřej Bojar, Rajen Chatterjee, Vishrav Chaudhary, Marta R. Costa-jussa, Cristina España-Bonet, Angela Fan, Christian Federmann, Markus Freitag, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Leonie Harter, Kenneth Heafield, Christopher Homan, Matthias Huck, Kwabena Amponsah-Kaakyire, Jungo Kasai, Daniel Khashabi, Kevin Knight, Tom Kocmi, Philipp Koehn, Nicholas Lourie, Christof Monz, Makoto Morishita, Masaaki Nagata, Ajay Nagesh, Toshiaki Nakazawa, Matteo Negri, Santanu Pal, Allahsera Auguste Tapo, Marco Turchi, Valentin Vydrin, and Marcos Zampieri. 2021. Findings of the 2021 Conference on Machine Translation (WMT21). In *Proceedings of the Sixth Conference on Machine Translation*, pages 1–88, Online. Association for Computational Linguistics.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Barry Haddow, Matthias Huck, Chris Hokamp, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Carolina Scarton, Lucia Specia, and Marco Turchi. 2015. Findings of the 2015 Workshop on Statistical Machine Translation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 1–46, Lisbon, Portugal. Association for Computational Linguistics.

Rajen Chatterjee, Christian Federmann, Matteo Negri, and Marco Turchi. 2019. Findings of the WMT 2019 Shared Task on Automatic Post-Editing. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 11–28, Florence, Italy. Association for Computational Linguistics.

Rajen Chatterjee, Markus Freitag, Matteo Negri, and Marco Turchi. 2020. Findings of the WMT 2020 Shared Task on Automatic Post-Editing. In *Proceedings of the Fifth Conference on Machine Translation*, pages 646–659, Online. Association for Computational Linguistics.

Rajen Chatterjee, Matteo Negri, Raphael Rubino, and Marco Turchi. 2018. Findings of the WMT 2018 Shared Task on Automatic Post-Editing. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 710–725, Belgium, Brussels. Association for Computational Linguistics.

Jonathan H. Clark, Chris Dyer, Alon Lavie, and Noah A. Smith. 2011. Better Hypothesis Testing for Statistical Machine Translation: Controlling for Optimizer Instability. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 176–181, Portland, Oregon, USA. Association for Computational Linguistics.

Tilman N. Höhle. 2019. Topologische Felder. In Stefan Müller, Marga Reis, and Frank Richter, editors, *Beiträge zur deutschen Grammatik: Gesammelte Schriften von Tilman N. Höhle*, 2 edition, volume 5 of *Classics in Linguistics*, pages 7–90. Language Science Press, Berlin, Germany.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. OpenNMT: Open-Source Toolkit for Neural Machine Translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, Vancouver, Canada. Association for Computational Linguistics.

Kevin Knight and Ishwar Chander. 1994. Automated Postediting of Documents. In *Proceedings of the AAAI Conference on Artificial Intelligence, 12*, pages 779–784.

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 Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Tom M. Mitchell. 1980. The Need for Biases in Learning Generalizations. Technical report, Rutgers University, New Brunswick, NJ.

Matteo Negri, Marco Turchi, Rajen Chatterjee, and Nicola Bertoldi. 2018. eSCAPE: a Large-scale Synthetic Corpus for Automatic Post-Editing. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, pages 24–30, Miyazaki, Japan. European Language Resources Association (ELRA).

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*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Nikolaos Pappas, Lesly Miculicich, and James Henderson. 2018. Beyond Weight Tying: Learning Joint Input-Output Embeddings for Neural Machine Translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 73–83, Brussels, Belgium. Association for Computational Linguistics.

Marga Reis. 1980. On Justifying Topological Frames : 'Positional Field' and the Order of Nonverbal Constituents in German [0]. *Documentation et Recherche en Linguistique Allemande Vincennes*, 22-23:59–85.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. 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)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Jaehun Shin, Wonkee Lee, Byung-Hyun Go, Baikjin Jung, Youngkil Kim, and Jong-Hyeok Lee. 2021. Exploration of Effective Attention Strategies for Neural Automatic Post-Editing with Transformer. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 20(6).

Matthew Snover, Bonnie Dorr, Rich Schwartz, Linnea Micciulla, and John Makhoul. 2006. A Study of Translation Edit Rate with Targeted Human Annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 223–231, Cambridge, Massachusetts, USA. Association for Machine Translation in the Americas.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to Sequence Learning with Neural Networks. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.

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 *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Angelika Wöllstein. 2018. Topologisches Satzmodell. In Jörg Hagemann and Sven Staffeldt, editors, *Syntaxtheorien: Analysen im Vergleich*, volume 28 of *Stauffenburg Einführungen*, pages 145–166. Stauffenburg, Tübingen, Germany.

Yilin Yang, Liang Huang, and Mingbo Ma. 2018. Breaking the Beam Search Curse: A Study of (Re-)Scoring Methods and Stopping Criteria for Neural Machine Translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3054–3059, Brussels, Belgium. Association for Computational Linguistics.

A Experimental Details

We use the following hyperparameters: the number of layers $N = 6$, the number of heads $H = 8$, the dimension of key vectors $d_k = 64$, the dimension of value vectors $d_v = 64$, the vector dimension for multi-head attention layers $d_{\text{model}} = 512$, the vector dimension for the inner layer of position-wise feed-forward networks $d_{\text{ff}} = 2{,}048$, the dropout (Srivastava et al., 2014) probability $P_{\text{drop}} = 0.1$, the label smoothing value $\epsilon_{\text{LS}} = 0.1$, minibatches of 25,000 tokens, a learning rate of 2.0, warmup for 18,000 training steps, and a shared vocabulary consisting of 32,000 subword units (Sennrich et al., 2016)[1]. We also use weight tying (Pappas et al., 2018) and the Adam optimizer (Kingma and Ba, 2015) with $\beta_1 = 0.9$, $\beta_2 = 0.998$, and $\epsilon = 10^{-8}$. Decoding options are beam search with a beam size $b = 5$, a length penalty multiplied by a strength coefficient $\alpha = 0.6$, and beam search stopping (Yang et al., 2018) with the length ratio $lr = 1.3$.

We use OpenNMT-py 3.0 (Klein et al., 2017)[2] with the random seed 1128. We first train the models for 100,000 steps, about 36 hours on one NVIDIA GeForce RTX™ 3090, and then tune them around 1,000 steps.

[1] We used SentencePiece (Apache License 2.0)

[2] The MIT License.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Section 8.

☒ A2. Did you discuss any potential risks of your work?
With the standard setup, studies in the field of automatic postediting, including this work, do not involve potential risks.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Section 1.

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Section 4 and Section 5.

☑ B1. Did you cite the creators of artifacts you used?
Section 4 and Section 5.

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Section 4, Section 5, and Appendix A.

☒ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
The used artifacts are not considered to have any extraordinary usages.

☒ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
The used artifacts are not considered to contain any information that names or uniquely identifies individual people or offensive content.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Section 1 and Section 4.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Section 4 and Section 5.

C ☑ Did you run computational experiments?

Section 4.

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Appendix A. The number of parameters are reported in the study on the baseline system.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Appendix A.

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 5.

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Section 4, Section 5, and Appendix A.

D ☑ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Section 5. However, the person who helped us only double-checked our analysis; he did not annotate any data.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Not applicable. Left blank.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Not applicable. Left blank.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Not applicable. Left blank.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.