Adaptation of Back-translation to Automatic Post-Editing for Synthetic Data Generation

WonKee Lee[†*]   **Baikjin Jung**[†*]   **Jaehun Shin**[†]   **Jong-Hyeok Lee**[†‡]

[†]Departmet of Computer Science and Engineering,
[‡]Graduate School of Artificial Intelligence,
Pohang University of Science and Technology (POSTECH), Republic of Korea
{wklee, bjjung, jaehun.shin, jhlee}@postech.ac.kr

Abstract

Automatic Post-Editing (APE) aims to correct errors in the output of a given machine translation (MT) system. Although data-driven approaches have become prevalent also in the APE task as in many other NLP tasks, there has been a lack of qualified training data due to the high cost of manual construction. eSCAPE, a synthetic APE corpus, has been widely used to alleviate the data scarcity, but it might not address genuine APE corpora's characteristic that the post-edited sentence should be a minimally edited revision of the given MT output. Therefore, we propose two new methods of synthesizing additional MT outputs by adapting back-translation to the APE task, obtaining robust enlargements of the existing synthetic APE training dataset[1]. Experimental results on the WMT English-German APE benchmarks demonstrate that our enlarged datasets are effective in improving APE performance.

1 Introduction

Automatic Post-Editing (APE) seeks to automatically correct errors included in the output of a black-box machine translation (MT) system to improve the final translation quality, thereby reducing the effort required for manual post-editing (Allen and Hogan, 2000; Chatterjee et al., 2015; Bojar et al., 2016; Chatterjee et al., 2018). In general, APE can be considered as a task of sequence-to-sequence supervised learning, which requires a considerable amount of human-annotated data. However, constructing an APE corpus—a set of triplets (Table 1), each of which includes a source text (*src*), a machine-translated text (*mt*), and a manually post-edited text (*pe*)—is labor-intensive work because

---

* Equal contribution to this work.
[1]Our synthetic APE data is available at https://github.com/wonkeelee/APE-backtranslation.git

src	Manipulates the shape of an item .
mt	**Bearbeitet** die Form eines Elements **an** .
pe	**Verändert** die Form eines Elements .

Table 1: An example of APE triplets from the WMT dataset (Bojar et al., 2017). **Boldface** words are either incorrect words in *mt* or post-edited words in *pe*.

post-editors should create *pe* in principle by **minimally editing** *mt* while preserving the meaning of *src*. In fact, the sizes of currently available 'genuine' APE corpora provided by WMT (Bojar et al., 2016, 2017; Chatterjee et al., 2018, 2019, 2020) are too small to train deep APE models effectively.

To overcome the lack of genuine APE corpora, several previous studies have proposed methods to construct synthetic training datasets (Junczys-Dowmunt and Grundkiewicz, 2016; Negri et al., 2018; Lee et al., 2020), and they appear to be partially helpful in mitigating the data scarcity problem. One such study is eSCAPE (Negri et al., 2018), which has been shown to be effective in training deep models and adopted in a number of APE works (do Carmo et al., 2020). Utilizing parallel corpora, which comprise pairs of a source sentence (*src*) and a reference sentence (*ref*), eSCAPE was constructed as a set of synthetic APE triplets in the form of (*src*, *mt*, *ref*) where *mt* is a machine translation of *src*, and *ref* serves as an alternative to *pe* of a genuine APE triplet.

Despite the effectiveness of eSCAPE, we argue that it may have two major drawbacks: (1) eSCAPE's method relies heavily on parallel resources, so its scalability is restricted to the quantity of available parallel resources and can be even more limited in low-resource scenarios; (2) the relation between *mt* and *ref* may not thoroughly reflect the actual relation between *mt* and *pe* because *ref* is not guaranteed to be a minimally edited revision of *mt*, potentially leading to the discrepancy between
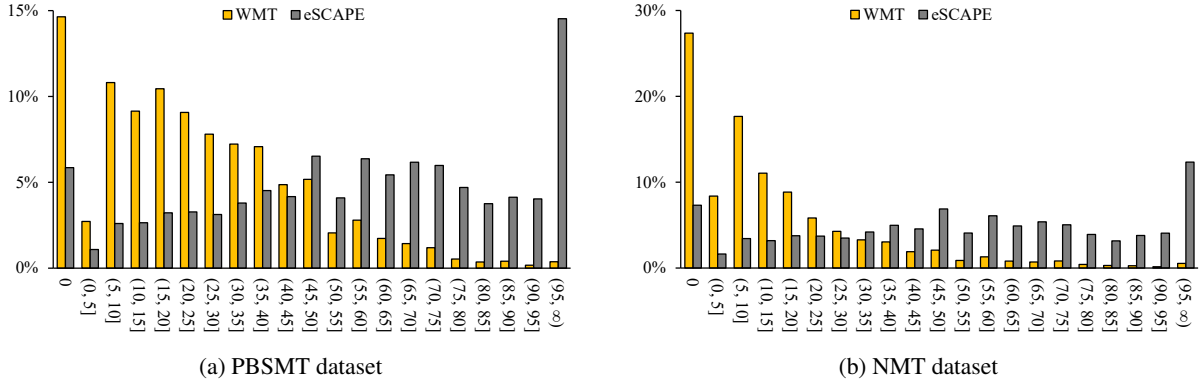
Figure 1: (a) and (b) are categorical distributions that indicate the proportion y [%] of samples belonging to a specific TER interval x for each of the two APE data types classified according to the type of the target MT system. Here, TER is a metric that presents the distance (error ratio) between *mt* and its target (*pe* or *ref*).

the distribution of translation errors in genuine data and that of translation errors in the synthetic data (Figure 1).

In this paper, we propose two automatic methods that are inspired by back-translation from the MT task (Sennrich et al., 2016), employing the APE process in the forward direction and the backward direction with applying the eSCAPE resource to them to create additional synthetic *mt*. Our approach not only extends the existing resource, but also aims to better simulate the characteristics of real APE data by making our synthetic *mt* better approximate the error distribution of the WMT APE benchmark dataset.

2 Background and Related Work

Back-translation. Back-translation is a method to create synthetic source texts from clean target texts by using an MT system that is trained in the target–to–source direction. Back-translation has allowed many MT studies to use monolingual data to generate additional parallel data so that they alleviate data scarcity; moreover, it has also been successfully adopted by other NLP tasks such as summarization (Parida and Motlicek, 2019; Jernite, 2019) and grammatical error correction (Xie et al., 2018).

Learning Objective of APE. Given that APE aims to revise *mt* to *pe* while preserving the meaning of *src*, each one of the two sources (*src*, *mt*) plays a distinct and critical role: *src* is treated as an auxiliary source, not only offering intact semantic and contextual information but also being helpful in identifying mistranslation; *mt*, meanwhile, serves as the primary source, which needs to be
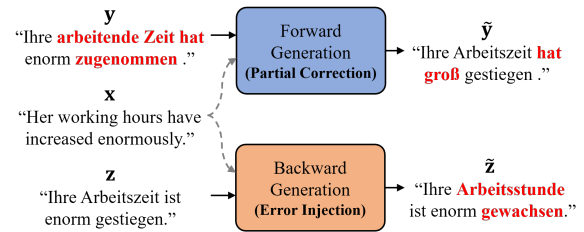

Figure 2: An illustration of our two synthetic data generation methods. **x**, **y**, and **z** are examples of *src*, *mt*, and *ref*, respectively. $\tilde{\mathbf{y}}$ and $\tilde{\mathbf{z}}$ are expected outputs that the proposed methods will produce. **Boldface** words indicate incorrect words.

corrected. In this perspective, the multi-source approach: (*src*, *mt*) $\mapsto$ *ref*, is commonly used to take both *src* and *mt* into account (Chatterjee et al., 2018, 2019). Specifically, considering *src*, *mt*, and *pe* as $\mathbf{x} = \{x_i\}_{i=1}^{T_x}$, $\mathbf{y} = \{y_j\}_{j=1}^{T_y}$, and $\mathbf{z} = \{z_k\}_{k=1}^{T_z}$ with the sequence lengths T_x, T_y, and T_z, respectively, the APE model learns to predict *pe* with the following conditional probability:

$$p(\mathbf{z}) = \prod_{k=1}^{T_z} p(z_k | \mathbf{x}, \mathbf{y}, z_{<k}; \theta), \qquad (1)$$

where θ is a set of model parameters.

3 Method

Beyond the eSCAPE corpus, to yield a more convincing error distribution as well as to supply APE models with more APE resources made out of limited parallel resources, we propose synthetic-data generation methods that can be seen as adaptations of back-translation to the APE task in terms of creating synthetic *mt*, which is one of the two sources of APE.

3686

We produce new synthetic *mt* so that *ref* can better act as its minimally post-edited text, whereas this *ref* may not do so for the original *mt*. Specifically, we suggest two strategies, both of which apply the APE process: '**forward generation**' and '**backward generation**'; each one of them performs APE in the forward direction and the backward direction, respectively. As described in Figure 2, the former partially corrects *mt* to reduce the distance between *mt* and *ref*, while the latter injects the right quantity of translation errors into *ref*.

3.1 Forward Generation

The 'Forward Generation' (FG) method lets an APE model take *src* and *mt* as input to produce mt_{FG} as output by partially correcting *mt* through the forward path of APE; the training objective of an FG model is identical to that of a normal APE model (Eq. 1). The output mt_{FG} then forms a new synthetic triplet (*src*, mt_{FG}, *ref*) together with *src* and *ref*. We use such triplets to construct a new set of synthetic triplets eSCAPE$_{\text{FG}}$.

Considering that *mt* generally requires a lot of excessive correction to match *ref*, this approach's motivation is that mt_{FG}, in itself a product of the APE process, will generally be closer to *ref* than the original *mt*. However, if the distance between mt_{FG} and *ref* is excessively small, indicating that the two texts are almost identical, APE models trained on eSCAPE$_{\text{FG}}$ may not learn error-correction patterns sufficiently. Thus, unlike the standard training procedure, we force the FG model's training process to stop earlier before convergence, making the remaining errors in its output mt_{FG} ample. We therefore use simple arrangements (§4) to find one optimal value for this stop point.

3.2 Backward Generation

Borrowing the idea of back-translation, the 'Backward Generation' (BG) method reverses the APE process during training by moving *mt* to the position of *ref* and vice versa; hence, a BG model is trained on (*src*,*ref*) $\mapsto$ *mt* to maximize the following conditional probability:

$$p(\mathbf{y}) = \prod_{j=1}^{T_y} p(y_j|\mathbf{x}, \mathbf{z}, y_{<j}; \theta). \quad (2)$$

In other words, the model learns to generate mt_{BG} to contain translation errors that occur in *mt* conditioned on a pair of *src* and *ref*. The output mt_{BG} then composes a new synthetic triplet (*src*,

Dataset		# Triplets
WMT-PBSMT	Train	23,000
	Dev	1,000
	Test2016	2,000
	Test2017	2,000
	Test2018	2,000
WMT-NMT	Train	13,442
	Dev	1,000
	Test2018	1,023
eSCAPE-PBSMT		7,258,533
eSCAPE-NMT		7,258,533

Table 2: Statistics of the WMT and eSCAPE datasets on the PBSMT and NMT subtasks.

mt_{BG}, *ref*) together with *src* and *ref*. We use such triplets to construct another set of synthetic triplets eSCAPE$_{\text{BG}}$.

In contrast to FG, the concept of BG is to corrupt a clean text (*ref*) by learning until the distance between the BG output and *ref* becomes similar to the edit distance of real APE data. However, if we let the BG model fully converge, the output mt_{BG} may not have a big difference from the original *mt*; on the other hand, if the model has been barely trained, mt_{BG} would be almost the same as *ref*. In both instances, APE models trained on eSCAPE$_{\text{BG}}$ may not learn error-correction patterns sufficiently. We use the same arrangements (§4) as in FG to find an optimal value for the BG model's stop point.

4 Experiments

Metric. Following the evaluation setting used in the WMT APE shared task, we adopt TER (Snover et al., 2006) as the primary metric to measure the distance between the model's prediction and the reference text; and BLEU (Koehn et al., 2007) as the secondary metric to measure the degree of n-gram match. In addition, all evaluations in our experiments are case-sensitive.

Dataset. We use two kinds of APE datasets: human-made APE datasets, which are provided by WMT, and eSCAPE. Both are English–German (EN–DE) APE corpora; they are further categorized according to their subtask depending on whether the target MT system is a phrase-based statistical MT (PBSMT) system or a neural MT (NMT) system. The WMT datasets are in the IT domain, whereas eSCAPE was made out of domain-general parallel corpora. Detailed data statistics are presented in Table 2. We tokenized all words in our datasets into sub-word units by using Senten-

	PBSMT								NMT	
MODELS	Test16		Test17		Test18		Avg.		Test18	
	TER(↓)	BLEU(↑)	TER(↓)	BLEU(↑)	TER(↓)	BLEU(↑)	TER(↓)	BLEU(↑)	TER(↓)	BLEU(↑)
WMT Baseline (No edit)	24.76	62.11	24.48	62.49	24.24	62.99	24.49	62.53	16.84	74.73
eSCAPE	16.97	73.94	17.35	72.93	17.74	72.34	17.35	73.07	16.39	75.70
eSCAPE$_{FG}$	17.06	73.96	17.40	72.81	18.00	72.19	17.48	72.98	16.30	75.77
eSCAPE$_{BG}$	17.25	73.58	17.85	72.30	17.93	72.12	17.66	72.66	16.50	75.40
eSCAPE + eSCAPE$_{FG}$	16.79	74.25	17.05*	73.30*	17.32*	72.95*	17.05*	73.50*	16.09*	76.11*
eSCAPE + eSCAPE$_{BG}$	16.73	74.32*	**16.96***	73.41*	17.26*	73.14*	16.98*	73.62*	**15.95***	**76.14***
eSCAPE + eSCAPE$_{FG}$ + eSCAPE$_{BG}$	**16.57***	**74.52***	16.99*	**73.50***	17.29*	73.11*	**16.95***	**73.71***	16.15*	76.00*
BERT-APE (Correia and Martins, 2019)	16.91	74.29	17.26	73.42	17.71	72.74	17.29	73.48	–	–
BERT-APE (Correia and Martins, 2019) (Ensemble)	16.49	74.98	16.83	73.94	17.15	73.60	16.82	74.17	–	–
BERT-APE (Lopes et al., 2019)	–	–	–	–	–	–	–	–	16.06	75.96

Table 3: Evaluation results of our APE models using different configurations on training datasets. '*' represents that our model's improvement is significant enough compared to the eSCAPE baseline in the second row with $p < 0.05$. The best result among our models in each column is in **bold** type. The three models at the bottom are current state-of-the-art models.

cePiece (Kudo and Richardson, 2018).

Model Configuration. We implemented a Transformer-based APE model, the "sequential" model proposed by Lee et al. (2019), which is one of the best performing models. We use this model both as generation models that create synthetic *mt* with our two proposed methods and also as the final APE models to examine the effectiveness of those synthesized data as additional training data. We follow the hyperparameter setting described in Lee et al. (2019), which again follows almost the same setting of the "base" Transformer described in the original paper Vaswani et al. (2017). However, we adjust the warm-up rate to 15,000 and the batch size to 25,000. We used OpenNMT-py[2] to implement and execute all models.

Synthetic Data Generation. To prevent our data generation model from generating what it has already seen during the training phase, we adopt the n-fold jack-knifing technique, which splits the whole dataset into $n - 1$ folds for training and 1 left-out fold for generation and validation, into our data generation process. Specifically,

1. Split eSCAPE into $n = 8$ folds: $\{f_i\}_{i=1}^8$.

2. Construct a training set,
   $\mathcal{D}_i = \text{Append}\left(\{f_j\}_{j=1}^8 \setminus \{f_i\}\right)$.

3. Train a data generation model (FG or BG) $\mathcal{M}_i$

on $\mathcal{D}_i$ and use 2,000 randomly extracted held-out samples from f_i for validation.

4. At a given model checkpoint, generate $\widetilde{mt}_i$ with $\mathcal{M}_i$ by supplying it with the pair of two sources in f_i.

5. Construct $mt_{FG/BG} = \text{Append}\left(\{\widetilde{mt}_i\}_{i=1}^8\right)$.

To examine the optimal stop point (§3.1, §3.2), we saved a model checkpoint every 25K training steps up to 150K steps, where the model converges with respect to its validation perplexity; thus, we obtained 6 sets of synthetic *mt* for each one of the two methods. Finally, for each method, we trained 6 APE models by using each new set of triplets including synthetic *mt*; and choose one set of synthetic *mt* that reports the best performance on the WMT validation dataset.

Evaluation With assistance from the FG and BG methods, we have a set of synthetic APE triplets $\mathcal{S} = \{\text{eSCAPE}, \text{eSCAPE}_{FG}, \text{eSCAPE}_{BG}\}$ available for training. In our experiments, we trained several APE models on various combinations of synthetic triplets in $\mathcal{S}$ together with the WMT training datasets and then compared the evaluation results to investigate how each data configuration affects the model's APE performance. Finally, after training the models until their perplexities on the WMT development dataset converge, we evaluated them on the WMT test datasets. We considered two baselines: (1) TER between *mt* and *pe* of the test datasets and (2) the performance of the APE model that is trained only on eSCAPE; the former

---

[2] https://github.com/OpenNMT/OpenNMT-py.git

Task	Gen. type	New samples (Ratio)	TER
PBSMT	FG	6,041,622 (83.23 %)	53.35
	BG	6,444,517 (88.79 %)	49.11
NMT	FG	4,969,521 (68.46 %)	52.53
	BG	6,304,471 (86.86 %)	45.89

Table 4: Statistics of synthetic *mt* produced by each proposed scheme. TERs are computed between *mt* and *ref*. 'New samples (Ratio)' indicates the number of synthetic *mt* that do not overlap with *mt* in eSCAPE.

implies that no post-editing has occurred yet, and it is used as the official baseline for the WMT APE shared task.

5 Results and Discussion

Table 3 shows the evaluation results. We observed that when eSCAPE$_{FG}$ or eSCAPE$_{BG}$ is used instead of eSCAPE, the APE model's performance does not make a big difference from the eSCAPE baseline. One possible reason that we expect is the gap between those synthetic *mt* and *mt* in the WMT dataset; in other words, synthetic *mt* is not produced by an existing MT system.

Nevertheless, we found that when we augment eSCAPE with eSCAPE$_{FG}$ and/or eSCAPE$_{BG}$, the trained APE model shows consistent improvements in its APE performance and most of the improvements upon the eSCAPE baseline are statistically significant. Moreover, the results also surpass current state-of-the-art (except the ensemble models) APE models (Correia and Martins, 2019; Lopes et al., 2019), which are built on top of BERT (Devlin et al., 2019), thus contain more model parameters, and exploit a huge amount of monolingual data. We expect that these results are because, in addition to an increase in the total quantity of training samples, the integration of multiple synthetic datasets, each of which focuses on different aspects of APE from the other—eSCAPE contains actual MT outputs; on the other hand, synthetic triplets better satisfy the minimal-edit criterion—appears to have an effect on the models' APE performance.

We found that our proposed methods derive a large number of new *mt* (Table 4) from eSCAPE and also yield a more similar TER distribution to that of WMT data than that of eSCAPE in terms of not only the mean TER (Table 4) but also the decrease in KL-divergence (Figure 3).
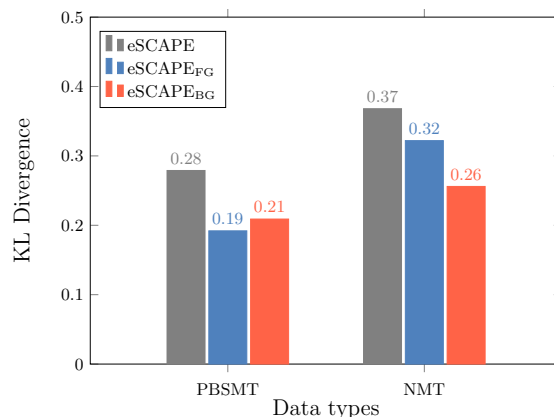

Figure 3: Presentation of $D_{KL}(P \parallel Q)$ (with base-10 logarithms) where P and Q are TER categorical distributions; P is for the WMT data, and Q is for each kind of synthetic triplets. The TER categorical distributions are plotted in the same manner as in Figure 1.

6 Conclusion and Future Work

In this paper, we tried to alleviate the drawbacks of eSCAPE by suggesting two new methods that adapt back-translation to the APE task, consequently increasing the data quantity and address the minimum editing characteristic. According to our experimental results, although APE models trained on each one of our two synthetic datasets show just comparable performances to the eSCAPE baseline, those trained on integrations of multiple synthetic datasets show consistent improvements over the baseline, implying that our new synthetic datasets are beneficial enlargements of eSCAPE. However, we manually selected the optimal stop points for both of our proposed generation schemes, so we will automate these selection processes in our future work.

Acknowledgments

References

Jeffrey Allen and Christopher Hogan. 2000. Toward the development of a post editing module for raw machine translation output: A controlled language

perspective. In *Proc. 3rd International Workshop on Controlled Language Applications (CLAW), 2000*, pages 62–71.

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

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurélie Névéol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. Findings of the 2016 Conference on Machine Translation. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 131–198, Berlin, Germany. Association for Computational Linguistics, Taberg Media Group AB.

Félix do Carmo, Dimitar Shterionov, Joss Moorkens, Joachim Wagner, Murhaf Hossari, Eric Paquin, Dag Schmidtke, Declan Groves, and Andy Way. 2020. A review of the state-of-the-art in automatic post-editing. *Machine Translation*, pages 1–43.

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.

Rajen Chatterjee, Marion Weller, Matteo Negri, and Marco Turchi. 2015. Exploring the planet of the APEs: a comparative study of state-of-the-art methods for MT automatic post-editing. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*

(Volume 2: Short Papers), pages 156–161, Beijing, China. Association for Computational Linguistics.

Gonçalo M. Correia and André F. T. Martins. 2019. A simple and effective approach to automatic post-editing with transfer learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3050–3056, Florence, Italy. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Yacine Jernite. 2019. Unsupervised Text Summarization via Mixed Model Back-Translation. arXiv:1908.08566 [cs.CL].

Marcin Junczys-Dowmunt and Roman Grundkiewicz. 2016. Log-linear Combinations of Monolingual and Bilingual Neural Machine Translation Models for Automatic Post-Editing. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 751–758, Berlin, Germany. 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 Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

WonKee Lee, Jaehun Shin, Baikjin Jung, Jihyung Lee, and Jong-Hyeok Lee. 2020. Noising Scheme for Data Augmentation in Automatic Post-Editing. In *Proceedings of the Fifth Conference on Machine Translation*, pages 783–788, Online. Association for Computational Linguistics.

WonKee Lee, Jaehun Shin, and Jong-Hyeok Lee. 2019. Transformer-based Automatic Post-Editing Model with Joint Encoder and Multi-source Attention of Decoder. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 112–117, Florence, Italy. Association for Computational Linguistics.

3690

António V. Lopes, M. Amin Farajian, Gonçalo M. Correia, Jonay Trénous, and André F. T. Martins. 2019. Unbabel's Submission to the WMT2019 APE Shared Task: BERT-Based Encoder-Decoder for Automatic Post-Editing. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 118–123, Florence, Italy. Association for Computational Linguistics.

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)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Shantipriya Parida and Petr Motlicek. 2019. Abstract Text Summarization: A Low Resource Challenge. 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 5994–5998, Hong Kong, China. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving Neural Machine Translation Models with Monolingual Data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Matthew Snover, Bonnie Dorr, Richard 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*, pages 223–231, Cambridge, Massachusetts, USA.

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*, pages 6000—-6010, Long Beach, California, USA. Curran Associates Inc.

Ziang Xie, Guillaume Genthial, Stanley Xie, Andrew Ng, and Dan Jurafsky. 2018. Noising and Denoising Natural Language: Diverse Backtranslation for Grammar Correction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 619–628, New Orleans, Louisiana. Association for Computational Linguistics.