Effort-Aware Neural Automatic Post-Editing

Amirhossein Tebbifakhr[1,2], **Matteo Negri**[1], **Marco Turchi**[1]
[1] Fondazione Bruno Kessler, Via Sommarive 18, Povo, Trento - Italy
[2] University of Trento, Italy
{atebbifakhr,negri,turchi}@fbk.eu

Abstract

For this round of the WMT 2019 APE shared task, our submission focuses on addressing the "over-correction" problem in APE. Over-correction occurs when the APE system tends to rephrase an already correct MT output, and the resulting sentence is penalized by a reference-based evaluation against human post-edits. Our intuition is that this problem can be prevented by informing the system about the predicted quality of the MT output or, in other terms, the expected amount of needed corrections. For this purpose, following the common approach in multilingual NMT, we prepend a special token to the beginning of both the source text and the MT output indicating the required amount of post-editing. Following the best submissions to the WMT 2018 APE shared task, our backbone architecture is based on multi-source Transformer to encode both the MT output and the corresponding source text. We participated both in the English-German and English-Russian subtasks. In the first subtask, our best submission improved the original MT output quality up to +0.98 BLEU and -0.47 TER. In the second subtask, where the higher quality of the MT output increases the risk of over-correction, none of our submitted runs was able to improve the MT output.

1 Introduction

Automatic Post-Editing (APE) is the task of correcting the possible errors in the output of a Machine Translation (MT) system. It is usually considered as a supervised sequence-to-sequence task, which aims to map the output of MT system to a better translation i.e. post-edited output, by leveraging a three-way parallel corpus containing (*source text*, *mt output*, *post-edited output*). Considering the MT output as a source sentence and the post-edited output as a target sentence, this problem can be cast as a monolingual translation task and be addressed with different MT solutions (Simard et al., 2007; Pal et al., 2016). However, it has been proven that better performance can be obtained by not only using the raw output of the MT system but also by leveraging the source text (Chatterjee et al., 2017). In the last round of the APE shared task (Chatterjee et al., 2018a), the top-ranked systems (Tebbifakhr et al., 2018; Junczys-Dowmunt and Grundkiewicz, 2018) were based on Transformer (Vaswani et al., 2017), the state-of-the-art architecture in neural MT (NMT), with two encoders to encode both source text and MT output. Although using these systems to post-edit the output of Phrase-Based Statistical Machine Translation (PBSMT) system resulted in a large boost in performance, smaller improvements were observed over neural MT outputs. Indeed, the good performance of the NMT systems leaves less room for improvement and poses the risk of over-correcting the MT output. Over-correction occurs when the APE system rephrases an already correct MT output. Although the post-edited output can still be a correct translation, it is penalized in terms of reference-based evaluation metrics, since it differs from the reference post-edited output.

With the steady improvement of NMT technology on the one side, and the adoption of reference-based evaluation metrics that penalizes correct but unnecessary corrections on the other side, tacking this problem has become a priority. In order to respond to this priority, for this round of the shared task our submission focuses on addressing the over-correction problem. Over-correction has been already addressed before by integrating Quality Estimation (QE) and APE system in three different ways (Chatterjee et al., 2018b), namely: *i)* as an *activator*, to decide whether to apply post-editing or not, using a threshold on the estimated

Proceedings of the Fourth Conference on Machine Translation (WMT), Volume 3: Shared Task Papers (Day 2) pages 139–144
Florence, Italy, August 1-2, 2019.

quality of the MT output, *ii)* as a *guidance*, to post-edit only the parts of a text that have poor estimated quality, *iii)* as a *selector*, to select the best output by comparing the estimated quality of the MT output and the automatically post-edited output. Our approach is a mixture of the first two. While in all previous scenarios the decision is made externally to the APE system, we allow the APE system to implicitly make the decision and in a softer manner. Instead of choosing between "*do*" and "*do not*" post-edit, we let the system decide which post-editing strategy to apply, choosing between three strategies: no post-editing (i.e. leaving the sentence untouched), light post-editing (i.e. a conservative modification) and heavy post-editing (i.e. an aggressive modification). To this aim, similar to the idea of multilingual NMT (Johnson et al., 2017), a special token is added to the beginning of both the source text and the MT output indicating the required amount of post-editing. Similar to last year's submission (Tebbifakhr et al., 2018), we use Transformer architecture with two encoders for encoding the source text and the MT output, while we share the parameters of the two encoders and tie the embeddings and decoder's softmax layer weights (Junczys-Dowmunt and Grundkiewicz, 2018).

We participated in both the APE sub-tasks proposed this year, which respectively consist in post-editing the output of English-German and English-Russian NMT systems. Our experiments show that, on the development sets for both language directions, prepending the special token can improve the performance of the APE system up to 0.5 BLEU points. However, predicting the correct token at test time, when the quality of the MT output is unknown, is still challenging and can harm the systems' performance. In the English-German subtask, our top system improves the MT output up to -0.47 TER and +0.98 BLEU points. In the English-Russian subtask, due to the high quality of the MT segments, none of our submitted systems was able to improve the MT output, emphasizing the need for further research towards more reliable solutions to the over-correction problem.

2 System Architecture

The backbone architecture of our system is based on the state-of-the-art architecture in NMT i.e. Transformer (Vaswani et al., 2017). Like most NMT models, it follows the encoder-decoder framework, where an encoder encodes the input sentence into a continuous space, and a decoder decodes this encoded representation into the output sentence. However, we use two encoders in order to process both the source text and the MT output. By attending to the concatenation of the representation of the source and MT sentences, the decoder generates the post-edited output. Following Junczys-Dowmunt and Grundkiewicz (2018), we share all the parameters between the encoders, and we use shared embedding weights across all encoders and the decoder and tie them to decoder's softmax layer weights.

In order to tackle the over-correction problem and to induce a post-editing strategy that resembles the work of a human post-editor, we add a special token to the beginning of both the source text and the MT output indicating the amount of required post-editing. In this paper, we use three different tokens, namely "*no post-edit*" (no edits are required), "*light post-edit*" (minimal edits are required), and "*heavy post-edit*" (a large number of edits are required). However, the number of tokens can be increased/decreased to provide more fine/coarse-grained information to the APE system, but this is beyond the scope of this paper. Before training, we first compute the TER (Snover et al., 2006) score between the MT output and the post-edited output, then we add the *no post-edit* token to samples with zero TER score, *light post-edit* to samples with non-zero TER score smaller than 40, and finally *heavy post-edit* to samples with TER score larger than 40. According to (Turchi et al., 2013, 2014), 40 TER is the level of quality above which a human translator tends to rewrite the post-edited sentence from scratch.

At testing time, since the post-edited output is not available, we need to predict the proper token for the input sample. For predicting the proper token, we test two approaches. The first one, namely BERT, is based on a text classifier obtained by fine-tuning BERT (Devlin et al., 2018) on the in-domain data, which classifies the MT output into the three defined classes. The second one, namely SIM, is an information retrieval approach, that, given a query containing the source and the MT sentence to be post-edited, retrieves the most similar triplet (source, MT sentence and post-edit) from the training data using an inverted index. Then, similarly to (Farajian et al., 2017), the retrieved triplets are ranked based on the aver-

age of the sentence-level BLEU scores (Chen and Cherry, 2014) between *a)* the source segment in the query and the retrieved source sentence and *b)* the MT segment in the query and the retrieved MT sentence. For the most similar triplet, the TER between the MT sentence and the post-edit is computed and the token created. For highly repetitive and homogeneous corpora, the similarity between the top retrieved triplet and the query is quite high, but this is not always the case. So, to limit the risk of assigning a token obtained from the top triplet, but with a low similarity, a threshold (τ) is set. If the average sentence-level BLEU of the top retrieved triplet is above τ, the relative token is associated to the query, otherwise the most frequent token in the training data is used. Once the token is obtained, it is added to the source and the sentence to be post-edited during inference.

3 Experimental Settings

3.1 Data

The official training data of the APE shared task contains a small amount of in-domain data, in which the post-edited outputs are real human post-edits. To overcome the lack of data and to train neural APE models, the organizers also provided a large amount of synthetic data. For the En-Ru subtask, they provided the eSCAPE dataset (Negri et al., 2018), which is produced from a parallel corpus by considering the target sentences as artificial human post-edits and machine-translated source sentences as MT output. For the En-De subtask, in addition to the eSCAPE dataset, another synthetic dataset was made available, which is created using round-trip translation from a German monolingual corpus (Junczys-Dowmunt and Grundkiewicz, 2016). We clean the English to German/Russian eSCAPE dataset by removing *i)* samples with a length ratio between source text and post-edited output which is too different than the average and *ii)* samples where the source text language is not English or post-edited output language is not German/Russian. In order to reduce the vocabulary size, we apply Byte Pair Encoding (BPE) (Sennrich et al., 2016). We learn the BPE merging rules on the union of the source text, MT output and post-edit output to obtain a shared vocabulary.

3.2 Hyperparameters

In our APE system, we use 32K merging rules for applying BPE. We employ OpenNMT-tf toolkit (Klein et al., 2017) to implement our system. We use 512 dimensions for the word embedding and 6 layers for both the encoders and the decoder, each containing 512 units and a feed-forward network with 1,024 dimensions. We set the attention and residual dropout probabilities, as well as the label-smoothing parameter to 0.1. For training the system, we use Adam optimizer (Kingma and Ba, 2014) with effective batch size of 8,192 tokens and the warm-up strategy introduced by (Vaswani et al., 2017) with warm-up steps equal to 8,000. We also employ beam search with beam width of 4.

3.3 Evaluation Metrics

We use two different evaluation metrics to assess the quality of our APE systems: *i)* TER (Snover et al., 2006), the official metric for the task, computed based on the edit distance between the given hypothesis and the reference and *ii)* BLEU (Papineni et al., 2002), as the geometric average of $n-$gram precisions in the given hypothesis multiplied by the brevity penalty.

4 Results

For both subtasks, we train our APE systems with and without prepending the token. We start the training of the APE systems on the union of the synthetic data and 20-times over-sampled in-domain data. Then, we fine-tune the best performing checkpoint on the development set only on the in-domain data. The best performance on the development sets for En-De and En-Ru is reported in Tables 1 and 2 respectively.

As shown in Table 1, both APE systems, with the oracle token and without the token (lines 2 and 3), improve the quality of the MT output for En-De subtask. This improvement is larger when the token indicating the required amount of post-editing is provided to the system. This observation confirms the need for guiding the APE system to adopt different post-editing strategies according to the MT quality. For the En-Ru subtask, as shown in line 2 and 3 of Table 2, although none of the two systems can improve over the MT output, the system with the token has better performance compared to the one without. However, during testing, the oracle token is not available and

Systems	TER (↓)	BLEU (↑)
MT Output	15.08	76.76
Without Token	14.65	77.55
Token (ORACLE)	14.38	77.85
Token (BERT)	15.54	76.56
Token (SIM)	15.31	77.06
Robust (BERT)	15.04	77.24
Robust (SIM)	15.07	77.24

Table 1: Performance of the APE systems, on the English-German development set.

Systems	TER (↓)	BLEU (↑)
MT Output	13.12	79.97
Without Token	14.92	78.17
Token (ORACLE)	14.77	78.51
Token (BERT)	15.72	77.28
Token (SIM)	15.07	77.97
Robust (BERT)	15.85	77.19
Robust (SIM)	15.04	78.09

Table 2: Performance of the APE systems, on the English-Russian development set.

we need to predict the proper token for each input sample. We run our post-editing system using the predicted tokens obtained by the approach based on the BERT text classifier (BERT) and the information retrieval method (SIM). [1] As reported in the lower part of both tables, performance drops when the predicted tokens are prepended to the source text and the MT output instead of the oracle tokens. On the one side, this shows that the errors made by our predicting approaches hurt the work of the APE. On the other side, this drop in performance confirms that the APE system is able to leverage the token when generating the post-edited output. In order to make the APE robust to the wrong token, we run the fine-tuning step on in-domain data using noisy tokens instead of oracle ones. To add noise to the tokens, we replace 30 percent of the tokens in the in-domain train data with a different token, randomly sampled from the two wrong labels. As shown in the last two lines of each table, adding noise to the tokens during training improves the results. In En-De, both approaches (BERT and SIM) have similar performance, while in En-Ru, the approach based on retrieving similar samples outperforms the approach using the text classifier. This is due to the fact that in En-Ru the majority token is "no post-edit" and the information retrieval approach tends to choose the majority token when the similarity is above the threshold resulting in more conservative post-editing. We submitted our best performing system without prepending the token as our *Primary* submission, and the best robust system with predicted tokens using the retrieval approach as our *Contrastive* submission. The results on English-German and English-Russian test sets are reported in Tables 3 and 4 respectively. These results confirm our findings on the dev data showing that *i)* the APE system is not able to improve the quality of the baseline for En-Ru, while it has limited gains for En-De and *ii)* the addition of the token seems to be more useful for En-Ru than for En-De, resulting in a small gain in BLEU compared to the system without prepending the token.

[1] The most frequent label in the En-Ru in-domain dataset is "*no post-edit*", while for En-De is "*light post-edit*". The τ values are 0.75 for En-Ru and 0.5 for En-De.

Systems	TER (↓)	BLEU (↑)
MT Output	16.84	74.73
Primary	16.37	75.71
Contrastive	16.61	75.28

Table 3: Performance of the APE systems, on the English-German test set.

Systems	TER (↓)	BLEU (↑)
MT Output	16.16	76.20
Primary	19.34	72.42
Contrastive	19.48	72.91

Table 4: Performance of the APE systems, on the English-Russian test set.

5 Conclusions

For this round of the APE shared task, we focused on the over-correction problem. In order to address this problem, we augmented the input of the APE system with a token to guide the system to be conservative when the MT output has high quality and aggressive with low-quality MT segments. Our experiments showed that it can result in bet-

ter performance when the added token is accurate. In fact, when the token has to be predicted during testing, it results in lower APE performance. In order to make the APE system robust to this noise, we fine-tune the APE system on in-domain data by altering a portion of the tokens in the data. This can help the system to be more robust against the noisy token at test time, but it still shows lower performance than the system without the token. We learned that it is necessary for the system to be aware of the quality of the MT output before applying the post-editing. However, predicting the quality of the MT output is still an open problem which has to be addressed.

References

Rajen Chatterjee, M. Amin Farajian, Matteo Negri, Marco Turchi, Ankit Srivastava, and Santanu Pal. 2017. Multi-source neural automatic post-editing: FBK's participation in the WMT 2017 APE shared task. In *Proceedings of the Second Conference on Machine Translation*, pages 630–638, Copenhagen, Denmark. Association for Computational Linguistics.

Rajen Chatterjee, Matteo Negri, Raphael Rubino, and Marco Turchi. 2018a. 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, Matteo Negri, Marco Turchi, Frédéric Blain, and Lucia Specia. 2018b. Combining quality estimation and automatic post-editing to enhance machine translation output. In *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Papers)*, pages 26–38, Boston, MA. Association for Machine Translation in the Americas.

Boxing Chen and Colin Cherry. 2014. A systematic comparison of smoothing techniques for sentence-level bleu. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 362–367.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

M Amin Farajian, Marco Turchi, Matteo Negri, and Marcello Federico. 2017. Multi-domain neural machine translation through unsupervised adaptation. In *Proceedings of the Second Conference on Machine Translation*, pages 127–137.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

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. Association for Computational Linguistics.

Marcin Junczys-Dowmunt and Roman Grundkiewicz. 2018. MS-UEdin submission to the WMT2018 APE shared task: Dual-source transformer for automatic post-editing. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 822–826, Belgium, Brussels. Association for Computational Linguistics.

D. P. Kingma and J. Ba. 2014. Adam: A Method for Stochastic Optimization. *ArXiv e-prints*.

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.

M. Negri, M. Turchi, R. Chatterjee, and N. Bertoldi. 2018. eSCAPE: a Large-scale Synthetic Corpus for Automatic Post-Editing. *ArXiv e-prints*.

Santanu Pal, Sudip Kumar Naskar, Mihaela Vela, and Josef van Genabith. 2016. A neural network based approach to automatic post-editing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 281–286.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Philadelphia, Pennsylvania. Association for Computational Linguistics.

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.

Michel Simard, Nicola Ueffing, Pierre Isabelle, and Roland Kuhn. 2007. Rule-based translation with statistical phrase-based post-editing. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 203–206. 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 Association for Machine Translation in the Americas*, pages 223–231, Cambridge, Massachusetts, USA.

Amirhossein Tebbifakhr, Ruchit Agrawal, Matteo Negri, and Marco Turchi. 2018. Multi-source transformer with combined losses for automatic post editing. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 846–852, Belgium, Brussels. Association for Computational Linguistics.

Marco Turchi, Matteo Negri, and Marcello Federico. 2013. Coping with the subjectivity of human judgements in MT quality estimation. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 240–251, Sofia, Bulgaria. Association for Computational Linguistics.

Marco Turchi, Matteo Negri, and Marcello Federico. 2014. Data-driven annotation of binary MT quality estimation corpora based on human post-editions. *Machine Translation*, 28(3):281–308.

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 I Guyon, U V Luxburg, S Bengio, H Wallach, R Fergus, S Vishwanathan, and R Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.