# Alibaba's Submission for the WMT 2020 APE Shared Task: Improving Automatic Post-Editing with Pre-trained Conditional Cross-Lingual BERT

**Jiayi Wang***, **Ke Wang***, **Kai Fan, Yuqi Zhang, Jun Lu, Xin Ge, Yangbin Shi, Yu Zhao**
Alibaba Group Inc., Hangzhou, China
{joanne.wjy,moyu.wk,k.fan,chenwei.zyq,joelu.luj,
shiyi.gx,taiwu.syb}@alibaba-inc.com, kongyu@taobao.com

## Abstract

The goal of Automatic Post-Editing (APE) is basically to examine the automatic methods for correcting translation errors generated by an unknown machine translation (MT) system. This paper describes Alibaba's submissions to the WMT 2020 APE Shared Task for the English-German language pair. We design a two-stage training pipeline. First, a BERT-like cross-lingual language model is pre-trained by randomly masking target sentences alone. Then, an additional neural decoder on the top of the pre-trained model is jointly fine-tuned for the APE task. We also apply an imitation learning strategy to augment a reasonable amount of pseudo APE training data, potentially preventing the model to overfit on the limited real training data and boosting the performance on held-out data. To verify our proposed model and data augmentation, we examine our approach with the well-known benchmarking English-German dataset from the WMT 2017 APE task. The experiment results demonstrate that our system significantly outperforms all other baselines and achieves the state-of-the-art performance. The final results on the WMT 2020 test dataset show that our submission can achieve +5.56 BLEU and -4.57 TER with respect to the official MT baseline.

## 1 Introduction and Related Work

Even machines can approach and achieve parity with human translations (Hassan et al., 2018) empowered by a sequence-to-sequence fashion (Bahdanau et al., 2014; Vaswani et al., 2017), post-editing is still an important and necessary step in the translation process, especially in scenarios where extremely high-quality translation results are essentially required such as business legal documents, technical product guides, medicine instructions and so on. It is the process whereby humans amend machine-generated translation to achieve an acceptable final product. Translation crowdsourcing paradigm, computer assisted translation (CAT) thus comes into being as demanded, which includes a hybrid of machine translation and human post-editing to meet translation scenarios with different quality requirements accordingly for accuracy, clarity, fluency, and domain adaptation.

However, post-editing, while improving, that can match human understanding of meaning, nuance, tone, humor–the list goes on, it's often worth paying extra more. The time spent on translation mistake corrections by humans remains substantial to the extent (Läubli et al., 2013) so that it even occasionally offsets the efficiency gained from the neural machine translation (NMT) systems. In this paper, we explore automatic post-editing (APE) in a deep learning framework where a two-stage training pipeline is engaged. The goal of APE task is to examine automatic methods of correcting translation mistakes produced by a black-box machine translation engine to improve the MT results. Human efforts are correspondingly reduced in the later editing process (Läubli et al., 2013) if our APE system can approach human translations as much as possible.

Traditionally, APE is a supervised learning task, requiring sufficient training data in the triplet of source (SRC), machine translation (MT) and post editing (PE) that are usually expensively available. Due to the limited number of such APE data released officially in this year's APE tasks and the specific domain, Wikipedia, which is quite different from the previous years'(IT domain), we adopt an imitation learning to mine WMT corpora, eSCAPE (Negri et al., 2018), Opus Wikipedia corpus (Wołk and Marasek, 2014) and our own English-German corpus to augment APE training data. However, pseudo data strategy is far from enough to train the state-of-the-art APE system. Inspired by the

* indicates equal contribution.



*Proceedings of the 5th Conference on Machine Translation (WMT)*, pages 789–796
Online, November 19–20, 2020. 

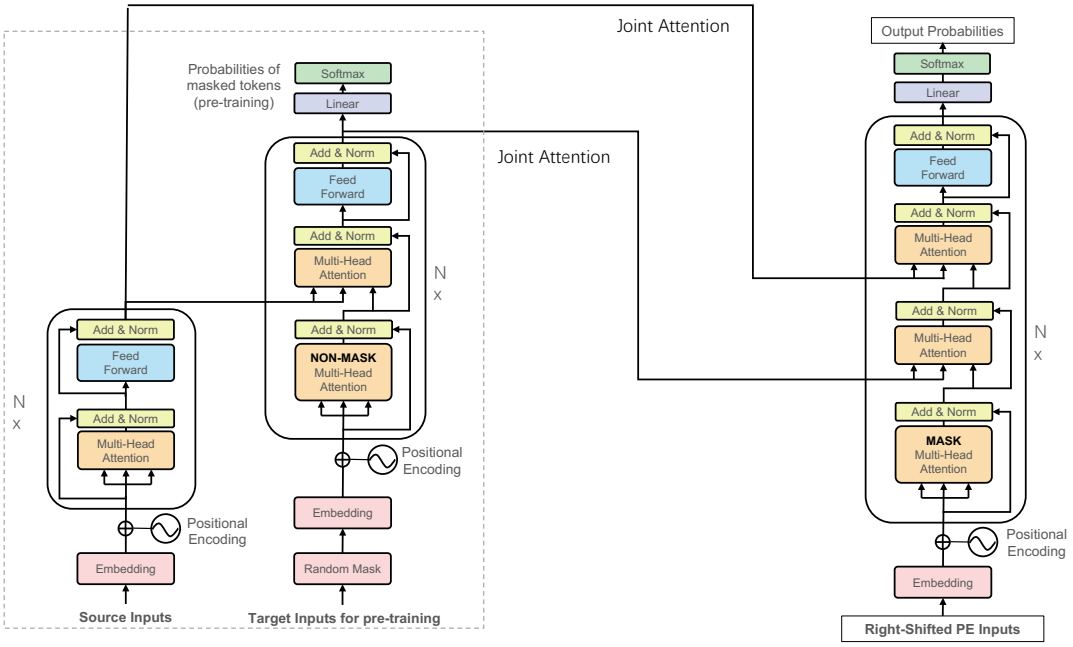



Figure 1: The APE model structure including detailed operations in pre-training and training

masked language model objective in the encoder BERT (Devlin et al., 2018), we introduce our Bert-like cross-lingual training objective to the encoder-decoder framework by adapting the decoder to become a memory encoder (Fan et al., 2019), allowing us to pre-train the target language model similar to BERT but conditioned on the source language text. Knowledge learned from the pre-training can be extensively transferred to many second-step downstream tasks, including but not limited to translation quality estimation, parallel corpus filtering and of course, automatic post-editing. The overall framework of our APE model is the same with the generative automatic post-editing model's structure in Wang et al. (2020).

Similar training mechanism is applied in the winner system of WMT 2019 APE Shared Task (Lopes et al., 2019), that wisely takes full advantage of the pre-trained multilingual BERT (mBERT) (Devlin et al., 2018) and achieves top performances.They concatenate the source and machine translation sentences to feed into the encoder mBERT and then fine tune the encoder and a transformer decoder where the context attention block is initialized by the self-attention weights of mBERT as well.

We examine our approach on the public English–German dataset from WMT 2017 APE shared task. Our system outperforms the top ranked methods in both BLEU (Papineni et al., 2002) and TER (Snover et al., 2006) metrics. For this year's WMT APE task, we finally submitted two ensemble English-German APE models according to different model selection methods and accomplish +5.56 increase in BLEU and -4.57 decline in TER on 2020 test set.

## 2 Methodology

In this section, we will introduce our APE model in terms of general structure and some computation details together with our data augmentation strategy.

### 2.1 APE Model Structure

The structure of our APE pre-training model originates from adapting the decoder in the transformer (Vaswani et al., 2017) to a memory encoder, following the exactly same design in Fan et al. (2019). We randomly pick 15% tokens in the target sentences during each training step to be substituted with a special [mask] token where predictions will be requisites accordingly, covering 12% masked, 1.5% substituted and 1.5% unchanged. In order to train a masked language model on the target sentences conditional on the sources, we accordingly remove the future mask matrix in the self-attention of the decoder to form a memory encoder, aiming to learn deep syntactic and alignment information of the ground truth. Therefore, during pre-training stage, the model is trained with high-quality English-German parallel corpora.

After we fully train the conditional language model on the target side, we apply an auto-regressive decoder on top of the pre-trained encoder-memory encoder model to decode the post-editing results in the stage of APE training by using the triplet data.

Figure 1 shows the details of our model structure. The part in the dotted line represents the pre-training stage with the removal of future mask in the memory encoder, and the whole picture describes the APE training process when the encoder-memory encoder pre-training model has been trained thoroughly. Note that the order of joint attentions of encoder and memory encoder with the decoder separately can be switched. Our experimental results in the following section illustrate this slight change can bring benefits to the diversity of the models and enhance the final ensemble's performance.

### 2.2 Data Strategies

**High-quality parallel corpus filtering** Our pre-training model requests high-quality parallel corpora. The dual conditional cross-entropy model (Junczys-Dowmunt, 2018) has been proven effective in WMT 2018 Corpus Filtering Shared Task. The cross-entropy scores according to two inverse translation models trained on clean data are used as the quality indicator so that we are able to mine qualified parallel sentences from noisy parallel corpora.

**APE training data augmentation.** Domain Adaption methods have been also investigated because of the small amount of official English-German APE training set and the special domain, Wikipedia. A semi-supervised CNN domain classification model (Chen and Huang, 2016) trained with in-domain seed and other general-domain data is utilized to extract in-domain source and target sentences from English-German corpora to augment pseudo sources and post-edits for APE training. To generate the corresponding machine translations of the classified in-domain source sentences, we use the rest of our corpus to train a neural machine translation model with model setting in Vaswani et al. (2017) to produce the MT results. The pseudo sources and post-edits are used as supplementary data during pre-training, and the pseudo triplets improve APE performance on the basis of only using official APE training set.

**Algorithm 1** Imitation Learning for Fine-tuning

**Require:** Reference Set $\mathbf{R} = \{(s_i, m_i, e_i)\}_{i=1}^{M}$, Full Training Set $\mathbf{T} = \{(s_j, m_j, e_j)\}_{j=1}^{N}$, hyperparameters $K \in [1, +\infty)$, $\alpha \in (0, 1)$.

Set the output dataset $\mathbf{R} = \{\}$.
**for** each $(\mathbf{s_i}, \mathbf{m_i}, \mathbf{e_i})$ in R **do**
  $\vec{V_r} = (TER(e_i, m_i), Length(e_i))$
  Candidate Set $C = \{\}$
  **for** each $(\mathbf{s_j}, \mathbf{m_j}, \mathbf{e_j})$ in T **do**
    $\vec{V_t} = (TER(e_j, m_j), Length(e_j))$
    **for** $m$ in 0,1 **do**
      **if** $\left\| (\vec{V_r}[m] - \vec{V_t}[m])/\vec{V_r}[m] \right\| > \alpha$ **then**
        Skip this training sample
      **end if**
    **end for**
    Add this training sample $(\mathbf{s_j}, \mathbf{m_j}, \mathbf{e_j})$ to $C$
  **end for**
  **if** size of $C > K$ **then**
    Sort candidates in $C$ by its cosine similarity to $\vec{V_r}$
    Remain only the top $K$ candidates in $C$
  **end if**
  **for** each candidate in $C$ **do**
    Add it to $F$
    Remove it from $T$
  **end for**
**end for**
**return** Filtered Dataset $F$.

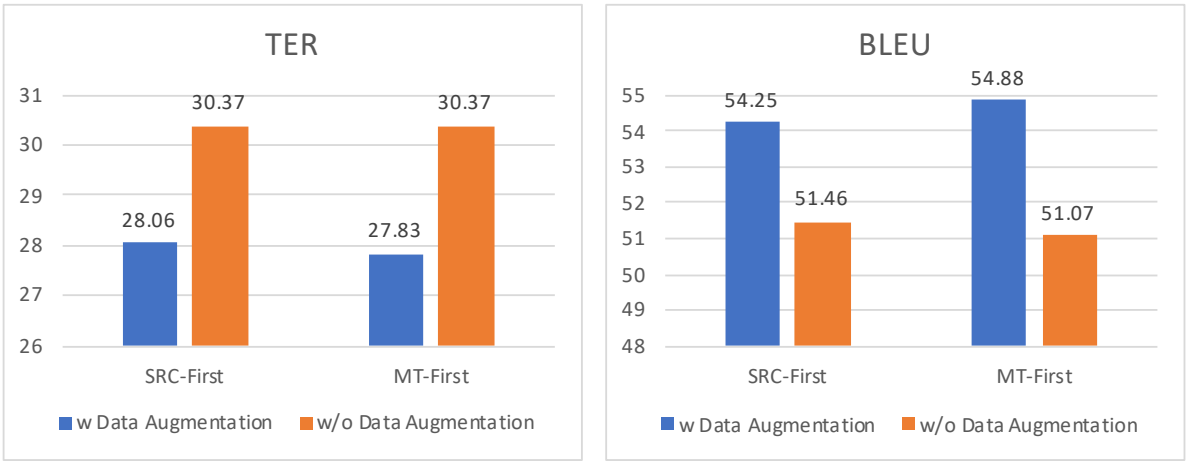
**Imitation learning.** To boost the APE model performance, we optimize our model during the APE training stage with further filtered APE data by an imitation learning method, since we noticed that there are gaps between the distributions of TERs in different types of our APE training set. Deeply motivated by Junczys-Dowmunt and Grundkiewicz (2016), we leverage the official training data containing real 7000 in-domain APE triplets as a reference set and apply Algorithm 1 to sample a subset of the whole training data in Table 1. Then we fine tune the APE model further with such a subset that has a similar distribution with this year's official training data. All the details of data usage will be described in the following experiment section.

## 3 Experiment

We conduct our experiments on two different datasets: First, to make a fair comparison with other top-ranked systems on WMT APE tasks in recent years, we perform a single model evaluation on the WMT 2017 English-German APE Shared Task without any other pseudo data except the Artificial dataset (Junczys-Dowmunt and Grundkiewicz, 2016) provided officially (for fair comparisons, and we avoid using the Escape Corpus (Negri et al., 2018) which has not been released until 2018); Second, we carry out a series of experiments

| Real/Pseudo | MT Engine | In/Out Domain | Up-sample Weight | Description | Size |
|---|---|---|---|---|---|
| Real | SMT | Out-domain | 10 | Train set of WMT 16&17 APE task | 23k |
| Real | NMT | Out-domain | 20 | Train set of WMT 18 APE task | 13.4k |
| Real | NMT | In-domain | 40 | Train set of WMT 20 APE task | 7k |
| Pseudo | SMT | Out-domain | 1 | Artificial Dataset | 4.4M |
| Pseudo | NMT | Out-domain | 1 | Escape Corpus (NMT) | 4.9M |
| Pseudo | NMT | In-domain | 1 | Our in-domain pseudo data | 20M |
| Total | - | - | - | Final training set | 30M |

Table 1: Compositions of the Training Data for the WMT 2020 APE Shared Task




(a) TER with/without Data Augmentation

(b) BLEU with/without Data Augmentation

Figure 2: Results in the English-German Development Set of WMT 2020 APE Shared Task of Different Model Structures with/without Data Augmentation

on the WMT 2020 English-German APE Shared Task with strategies including data argumentation, quality filtering, domain adaptation, and model ensemble to accomplish the overall performance of our model.

### 3.1 Setup

**Dataset.** For the experiments on WMT 2017 APE, we verify our APE model design on the open public WMT 2017 English-German APE Shared Task (Ondrej et al., 2017). The official training set consists of 23K real triplets (SRC, MT, PE) for training and another 2K triplets for testing from the Internet Technology (IT) domain. Besides, the shared task offers a large-scale artificial synthetic corpus containing around 500K high-quality and 4 million relatively low-quality synthetic triplets. We over sample the APE real data by 20 times and merge it with the synthetic data, resulting in roughly 5 million of triplets for both pre-training and APE training. The final APE system is selected based on WMT 2016 APE test set.

For the experiments on WMT 2020 APE, we use all available APE triplets of WMT English-German APE tasks released since 2016, including about 43.4K real triplets as well as 9.3M synthesized data made up with Artificial (Junczys-Dowmunt and Grundkiewicz, 2016) and Escape (Negri et al., 2018). Considering the application domain for this year's task changes from IT to Wikipedia and the size of the official in-domain training set is quite small (only 7000 samples), we generate about 20M in-domain pseudo data for our model training as follows:

1. We apply the cross-entropy scoring algorithm described in section 2.2 on our own English-German parallel corpus and filter out about 200 million high-quality parallel data with a proper threshold.

2. We collect the Wikipedia corpus from Wołk and Marasek (2014), which contains more than 2 million of English-German parallel sentences. We up-sample the SRC of this year's training data 20 times and mix them with the English side of Wikipedia corpus as our in-

domain seeds and train a domain classification model as described in section 2.2 with other general-domain data including the news and biomedical dataset from the WMT 2020 website. Afterwards, the domain classfication model is applied to extract about 20 million of in-domain parallel sentences from the 200M high-quality parallel data mentioned above.

3. The left 180 million are used to train a English-German transformer-based neural machine translation model (Vaswani et al., 2017) with the OpenNMT (Klein et al., 2017) source code. The sources and targets of the 20M high-quality in-domain parallel corpus are treated as SRCs and PEs and the decoding results from the trained NMT model are regarded as corresponding MTs. These in-domain pseudo triplets are mixed with all available training set from the WMT APE Shared Task since 2016 with differentiated up-sample weights as our final training set, as shown in Table 1.

**Pre-processing.** In all of our experiments, we apply truecasers trained independently for English and German seperately (Koehn et al., 2007) and process our data into subword units (Kudo, 2018) with a 32K shared vocabulary. Triplets with more than 70 subword units in any one of the SRCs, MTs or PEs are removed.

**Evaluation Metrics.** We mainly evaluate our systems with the metrics, translation edit rate (TER) (Snover et al., 2006) and bilingual evaluation understudy (BLEU) (Papineni et al., 2002), since they are standard and widely employed in evaluation of the WMT APE tasks.

**Model Setting.** All experiments are trained on 8 NVIDIA P100 GPUs for maximum 100,000 steps for about two days until convergence, with a total batch-size of 65536 tokens per step and the Adam optimizer (Kingma and Ba, 2014). Parameters are being tuned with 12,000 steps of learning rates warm-up (Vaswani et al., 2017). Except these modifications, we follow the default transformer-based configuration (Vaswani et al., 2017) for other hyper-parameters settings.

### 3.2 Results on WMT 2017 APE Shared Task

We verify the validity and efficiency of our proposed model on WMT 2017 APE test data since all of the winners of WMT APE Shared Tasks of recent years do report their results of single models on this dataset (Junczys-Dowmunt and Grundkiewicz, 2018; Correia and Martins, 2019). To make a fair comparison, we do not use any extra data for training as described in the data setup.

The main results of APE systems are presented in Table 2, demonstrating that our single model, even without pre-training, outperforms all winners of the WMT APE Shared Task from 2017 to 2019 on both BLEU and TER metrics.

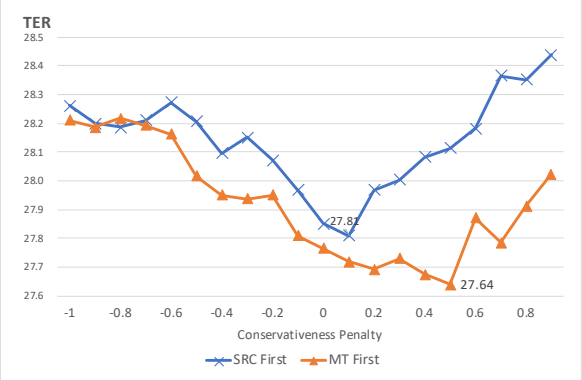



Figure 3: TERs on the English-German Development Set of WMT 2020 APE Shared Task from the Further Optimized Models with Different Values of Conservativeness Penalty

### 3.3 Results on WMT 2020 APE Shared Task

For this year's task, we adopt various of strategies including data augmentation, further optimization by imitation learning and model ensemble.

**Data Augmentation** As described in section 2.2, we utilize several algorithms, quality filtering and domain adaption, to construct our own in-domain pseudo data for APE training. We conduct experiments with and without in-domain pseudo data on two different model structures described in Section 2.1 for decoder joint attention switching (referred as SRC-First and MT-First respectively in the following discussion). Results on the 2020 development set in Figure 2 indicate that our data augmentation strategies can generate powerful pseudo data which significantly improve the model performance in this year's APE task.

**Further Optimization via Imitation Learning** The hyper-parameters $\alpha$ and $K$ in Algorithm 1 are set to 0.3 and 500 according to empirical studies. Finally, around 2M triplets are filtered from the full training set via the imitation learning algorithm. We compare TERs before and after APE fine tun-

| Model | BLEU↑ | TER↓ | Note |
|---|---|---|---|
| Official Baseline | 62.49 | 24.48 | Do nothing to the original machine translation |
| FBK (Ensemble) | 70.07 | 19.60 | Ensemble model, winner of WMT17 APE task |
| MS-UEdin | 69.72 | 19.49 | Single model, winner of WMT18 APE task |
| Unbabel (BED) | 70.66 | 19.03 | Single model, winner of WMT19 APE task. |
| Proposed Model w/o pre-training | 70.90 | 18.90 | Single model without pre-training |
| Proposed Model w pre-training | 71.52 | 18.44 | Single model with pre-training |

Table 2: Performance Comparisons on WMT 2017 APE English-German Test Set

| Model | BLEU↑ | TER↓ | Note |
|---|---|---|---|
| Official Baseline | 50.37 | 31.37 | Do nothing with the original machine translation |
| Ensemble×5 of BED | 55.09 | 27.85 | The winning system of last year |
| Our Single Model | 54.88 | 27.83 | MT-First structure |
| + Optimizing | 54.50 | 27.76 | Optimized on filtered subset |
| + Conservativeness Penalty | 54.87 | 27.64 | Conservativeness penalty = 0.5 |
| Our Ensemble×5 | 55.87 | 27.02 | Our contrastive submission |
| Our Ensemble×5 | **56.06** | **26.99** | Our primary submission |

Table 3: Main Results in the English-German Development Set of the WMT 2020 APE Shared Task

| Model | BLEU↑ | TER↓ |
|---|---|---|
| Official Baseline | 50.21 | 31.56 |
| Our Primary Submission | 55.58 | 27.03 |
| Our Contrastive Submission | **55.77** | **26.99** |

Table 4: Submission Results in the English-German Test Set of the WMT 2020 APE Shared Task

ing with the filtered data in Figure 4 with the two different model structures. It can be clearly shown that the APE model can be further improved.

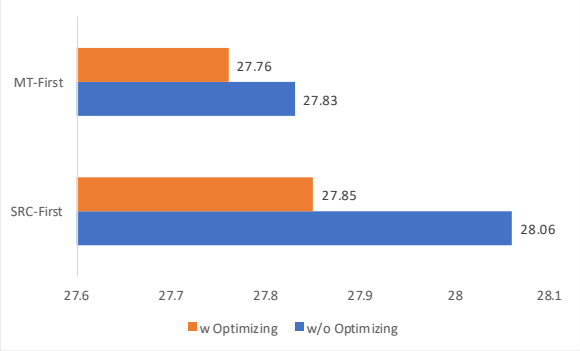



Figure 4: TERs on the English-German Development Set of WMT 2020 APE Shared Task for Different Model Structures with/without Further Optimizing

**Ensemble** We train the two different APE models (SRC-First & MT-First), each for three times with 30M APE training set to get 6 primary APE models and fine tune all of them with 2M filtered APE data via imitation learning for further optimization. Then, we obtain 12 APE models, 6 primary models and 6 optimized ones. Our final primary submission is an ensemble of the top 5 primary models with lowest TERs. In contrast, an ensemble of the top 5 optimized models is submitted as well for validation of imitation learning method.

Following the winning system of last year, we apply the conservativeness penalty (Lopes et al., 2019) on each model before ensemble. As shown in Figure 3, the local optimal solutions for the conservativeness penalty may be various among models. Therefore, instead of a fixed constant, we apply the most appropriate penalties for each model according to their performance on the 2020 development set. Results of our ensemble models in the development set and the test set can be found at Table 3 and Table 4 respectively.

Besides, we also train last year's winning system five times (BED (Lopes et al., 2019)) with the exactly same data we use for WMT 2020 APE task based on the source code they released[1] and pro-

[1]https://github.com/deep-spin/OpenNMT-APE

duce an ensemble result reported in Table 3. Evaluated on 2020 development set, both of our final ensemble model in primary and contrastive submissions outperform the winning system of 2019. The final results on the 2020 test set released officially show that our ensemble models significantly improve the machine translations with significant margins in TER and BLEU (-4.57 TER and +5.56 BLEU).

## 4 Conclusion

This paper describes our automatic post-editing system for the WMT 2020 English-German APE Shared Task. We introduce a cross-lingual Bert-like conditional model with an innovative memory encoder which can capture the deep semantic information of machine translations conditional on the source sentences. In addition, efforts on data augmentation strategies, corpus filtering and imitation learning, are able to overcome the scarcity of real APE data and further improve the model performance together with the ensemble strategy. Our single APE model outperforms all winner systems of recent years' WMT APE Shared Tasks on the WMT 2017 English-German test set and achieves impressive performances on the WMT 2020 English-German APE test set.

## Acknowledgments

This work is partly supported by National Key RD Program of China (2018YFB1403202).

## References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Boxing Chen and Fei Huang. 2016. Semi-supervised convolutional networks for translation adaptation with tiny amount of in-domain data. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 314–323.

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. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Kai Fan, Jiayi Wang, Bo Li, Boxing Chen, and Niyu Ge. 2019. Neural zero-inflated quality estimation model for automatic speech recognition system. *arXiv preprint arXiv:1910.01289*.

Hany Hassan, Anthony Aue, Chang Chen, Vishal Chowdhary, Jonathan Clark, Christian Federmann, Xuedong Huang, Marcin Junczys-Dowmunt, William Lewis, Mu Li, et al. 2018. Achieving human parity on automatic chinese to english news translation. *arXiv preprint arXiv:1803.05567*.

Marcin Junczys-Dowmunt. 2018. Dual conditional cross-entropy filtering of noisy parallel corpora. *arXiv preprint arXiv:1809.00197*.

Marcin Junczys-Dowmunt and Roman Grundkiewicz. 2016. Log-linear combinations of monolingual and bilingual neural machine translation models for automatic post-editing. *arXiv preprint arXiv:1605.04800*.

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, Volume 2: Shared Task Papers*, pages 835–839. Association for Computational Linguistics.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M Rush. 2017. Opennmt: Open-source toolkit for neural machine translation. *arXiv preprint arXiv:1701.02810*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 177–180. Association for Computational Linguistics.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. *arXiv preprint arXiv:1804.10959*.

Samuel Läubli, Mark Fishel, Gary Massey, Maureen Ehrensberger-Dow, Martin Volk, Sharon O'Brien, Michel Simard, and Lucia Specia. 2013. Assessing post-editing efficiency in a realistic translation environment.

António V Lopes, M Amin Farajian, Gonçalo M Correia, Jonay Trenous, and André FT Martins. 2019. Unbabel's submission to the wmt2019 ape shared task: Bert-based encoder-decoder for automatic post-editing. *arXiv preprint arXiv:1905.13068*.

Matteo Negri, Marco Turchi, Rajen Chatterjee, and Nicola Bertoldi. 2018. Escape: a large-scale synthetic corpus for automatic post-editing. *arXiv preprint arXiv:1803.07274*.

Bojar Ondrej, Rajen Chatterjee, Federmann Christian, Graham Yvette, Haddow Barry, Huck Matthias, Koehn Philipp, Liu Qun, Logacheva Varvara, Monz Christof, et al. 2017. Findings of the 2017 conference on machine translation (wmt17). In *Second Conference onMachine Translation*, pages 169–214. The Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

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*, volume 200. Cambridge, MA.

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*, pages 5998–6008.

Ke Wang, Jiayi Wang, Niyu Ge, Yangbing Shi, Yu Zhao, and Kai Fan. 2020. Computer assisted translation with neural quality estimation and automatic post-editing. *arXiv preprint arXiv:2009.09126*.

Krzysztof Wołk and Krzysztof Marasek. 2014. Building subject-aligned comparable corpora and mining it for truly parallel sentence pairs. *Procedia Technology*, 18:126–132.