Proceedings of the 13th Conference on Language Resources and Evaluation (LREC 2022), pages 883–891
Marseille, 20-25 June 2022

Empirical Analysis of Noising Scheme based Synthetic Data Generation for Automatic Post-editing

Hyeonseok Moon[1], Chanjun Park[1,2], Seolhwa Lee[3], Jaehyung Seo[1], Jungseob Lee[1], Sugyeong Eo[1], Heuiseok Lim[1,†]
[1]Korea University
Seoul, South Korea
[2]Upstage
Gyeonggi-do, South Korea
[3]University of Copenhagen
Copenhagen, Denmark
[1]{glee889, bcj1210, seojae777, omanma1928, djtnrud, limhseok}@korea.ac.kr
[2]chanjun.park@upstage.ai
[3]sele@di.ku.dk

Abstract

Automatic post-editing (APE) refers to a research field that aims to automatically correct errors included in the translation sentences derived by the machine translation system. This study has several limitations, considering the data acquisition, because there is no official dataset for most language pairs. Moreover, the amount of data is restricted even for language pairs in which official data has been released, such as WMT. To solve this problem and promote universal APE research regardless of APE data existence, this study proposes a method for automatically generating APE data based on a noising scheme from a parallel corpus. Particularly, we propose a human mimicking errors-based noising scheme that considers a practical correction process at the human level. We propose a precise inspection to attain high performance, and we derived the optimal noising schemes that show substantial effectiveness. Through these, we also demonstrate that depending on the type of noise, the noising scheme-based APE data generation may lead to inferior performance. In addition, we propose a dynamic noise injection strategy that enables the acquisition of a robust error correction capability and demonstrated its effectiveness by comparative analysis. This study enables obtaining a high performance APE model without human-generated data and can promote universal APE research for all language pairs targeting English.

Keywords: Machine Translation, Automatic Post-Editing, Noise Injection, Data Generation

1. Introduction

Automatic Post-editing (APE) aims at automatically correcting errors of translated sentences generated by machine translation systems (Chatterjee et al., 2020). The APE research field shows practical effectiveness as it can correct errors in translated sentence, and improve sentence quality regardless of the utilized machine translation system. Currently, it is being actively studied at various conferences such as Conference on Machine Translation (WMT) (Chatterjee et al., 2018). With the development of Neural Machine Translation (NMT) technology, APE has effectively reduced the human effort required to correct errors and has improved the quality of the translation significantly.

However, this study has a substantial limitation: time consumption and human resources that are required to generate the APE data (Negri et al., 2018). Generally, APE data comprises source sentences (SRC), its machine translation results (MT), and human post-edited sentences (PE) of the translation results. Through these data, the APE model is trained to generate PE by feeding SRC and MT as input (Chatterjee et al., 2020). The most challenging part is the generation of PE, which requires expert level human labor. More than simply translating a source sentence, deep inspection about the machine-like errors in a sentence must be figured out and properly corrected at the human level. This follows difficulties in the acquisition of training data. Considering WMT, the largest conference related to APE, only 7,000 and 1,000 training and validation/test data were released, respectively. Furthermore, this was only distributed for limited language pairs such as En-De and En-Zh. Most language pairs do not have officially released APE data (Chatterjee et al., 2019; Chatterjee et al., 2019). This remains a major restriction of universal research of APE for diverse language pairs.

Regarding the aforementioned limitation, some methods can be considered that only use a parallel corpus to generate APE data (Negri et al., 2018; Lee et al., 2021; Wang et al., 2020). Among them, we focus on ***generating APE data from a parallel corpus based on noise without human intervention***. Lee et al. (2020) proposed a noise generation method that considers the source sentence, target sentence of the parallel corpus as SRC, PE of the APE triplet (*i.e.,* SRC, *pseudo-*MT, PE), respectively. Specifically, they imposed the noise in PE to generate pseudo-MT(pMT).

† This author is the corresponding author.

This method can easily implement an APE triplet suitable for the user's purpose by applying different noise application ratios, and it can work effectively even in low-resource settings.

Despite the mentioned strengths, this noise-based data generation approach is substantially naive. Four types of noising-schemes were simply utilized grounded on edit distance: insertion, deletion, substitution, shifting [1]. Although the study improves practical performance, it is difficult to consider this noised data to be containing factual errors of machine translation systems that should be corrected at the human level (Lee et al., 2021).

In this study, we focus on resolving the previous approach's striking limitations by leveraging the noising scheme's benefit. This investigation is beyond the previous study, which is based on the edit distance as a noising strategy to ***leverage various noising schemes***. We propose standardized noise schemes mimicking the correction of errors at the human level, and derived optimized strategy through validating each noise scheme's effectiveness. These include noising schemes considering part-of-speech (Petrov et al., 2011) and semantic meaning (*i.e.,* synonym, thesaurus) for ***mimicking the human-level correction process***.

Furthermore, to achieve a higher performance through the noising scheme, we investigate the ***noise injection strategy***, which determines the application of noise in the training process. Specifically, we figured out two noise injection strategies, including static and dynamic noise injection. Applying the static noise injection, the APE training data are generated only once during the whole training process, whereas dynamic noise injection continuously generates APE data that have different noise for every training epoch. In this study, we show that dynamic noise injection can enable more robust and diverse acquisition of error correction capability. This study makes the following contributions.

- We propose APE model training methodologies that can be applied using only a parallel corpus. This obviates the need for expert human labor for the APE data generation and enables diverse APE model construction for all the language pairs targeting English.
- We also propose several noising schemes mimicking human-level correction process and have derived the optimal noising scheme that yields substantial effectiveness.
- We suggest practical training strategy by utilizing noising schemes in the APE model training and have verified the effectiveness of our proposed methods.

[1]Edit distance consists of insertion, deletion, substitution in general; nonetheless, in this study, we include 'shifting,' which is considered in calculating the translation edit rate (TER).

2. Related Studies

Studies to solve the problem of data shortage in APE are being actively conducted. Representatively, eSCAPE (Negri et al., 2018) synthesizes APE data form parallel data source based on a translation model. This method treats source and target sentences of a parallel corpus as SRC and PE of a APE triplet. Then by translating the SRC through the translation system, pMT that will serve as the MT of the APE training data, is generated. These studies work effectively on language pairs without the APE data and are actively used as a data augmentation scheme in previous APE studies, showing good performance (Lopes et al., 2019; Wang et al., 2020). However, as the translated sentences are independent of the PE, pMT does not properly reflect the errors that should be corrected at the human level (Lee et al., 2021).

To solve this problem, Lee et al. (2021) proposes the method of generating pMT similarly as back-translation. This method trains a model that generates MT with SRC and PE as inputs, and it generates pMT based on the model. This method can generate pMT to be containing practical errors that are corrected by human. Nonetheless, these training process requires human edited APE data for training the data generation model, and the quality of pMT is still highly dependent on the model performance.

For relieving such limitations, this study focuses on the noising scheme based data generation method (Lee et al., 2020). Through this method, we can obviate the necessity of translation model in data generation, and can generate pMT containing errors to be dealt with in an actual correction. Therefore, our study analyzes the weaknesses of the previous methodology and improves upon them, by adopting manifold noising schemes.

3. Proposed Method

3.1. Noise-based Data Generation

In this study, we propose new noising schemes mimicking errors in the translation system that is corrected at the human level. Particularly, we expand the existing noising schemes that are utilized in Moon et al. (2021a) and Lee et al. (2020) and propose various noising schemes based on the practical errors that need to be considered in the correction phase. These can easily be applied to all the language pairs that target English, regardless of its source language. Their universal applicability can promote extensive studies on various language pairs, especially for the language pairs that the APE data do not exist.

3.1.1. Edit Distance-based Noising Scheme

Noising scheme-based data generation is first proposed by Lee et al. (2020). Regarding the study, the APE data that comprise SRC, MT, and PE are generated by utilizing a parallel corpus. Source and target sentences in the parallel corpus are regarded as SRC and PE, respectively. Thereafter, the pseudo-MT(pMT) is syn-

thesized by injecting a pre-defined noising scheme to the PE.

Considering a previous study (Lee et al., 2020), edit distance, estimated in the evaluation of translation edit rate (TER) score (Snover et al., 2006), is utilized for the corresponding noising scheme. Specifically, the following four strategies are utilized: insertion (inserts random token to PE), deletion (deletes random tokens in the PE), substitution (changes tokens in the PE into random other tokens), and shifting (shuffles positions of the existing tokens in the PE). Regarding these methods, random tokens are randomly extracted from the training data. In this study, we denote these edit distance-based noising schemes as Ins_{ED} for the insertion, Del_{ED} for the deletion, Sub_{ED} for the substitution, and Shift_{ED} for the shifting.

The detailed process of injecting noising schemes is demonstrated below. Considering each sentence pair (SRC, TGT) in a parallel corpus D, we tokenize TGT into $\{tgt_i\}_{i=1}^{n}$, where n is the token length of TGT by utilizing natural language toolkit (NLTK) (Loper and Bird, 2002). We denote the bag of words which accumulate all the segmented tokens in TGT as L^{TGT}. Using the probability p, the noised sentence $pMT = \{pmt_i\}_{i=1}^{n}$ is generated as in Eq. (1):

$$pmt_i = \begin{cases} N_L(tgt_i) & \text{if } r \in [0, p) \\ tgt_i & \text{if } r \in [p, 1) \end{cases} \tag{1}$$

Considering this equation, r refers to the random variable extracted from the uniform distribution $U[0, 1)$ that determines the probability to be noised for each token. Following these notations, the edit distance-based noising schemes defined in above can be formularized as shown in Eq. (2):

$$N_L(tgt_i) = \begin{cases} tgt_i : l\,(l \in L) & : \text{Ins}_{\text{ED}} \\ \text{None} & : \text{Del}_{\text{ED}} \\ l\,(l \in L) & : \text{Sub}_{\text{ED}} \\ tgt_j\,(j \in [1, n]) & : \text{Shift}_{\text{ED}} \end{cases} \tag{2}$$
$$\text{where } L = \bigcup_{TGT \in D} L^{TGT}$$

By applying Shift_{ED}, pmt_j is simultaneously determined by tgt_i because pmt_i is replaced with $tgtj$. After the injection process, we construct the APE training triplet as (SRC, pMT, and PE) by regarding TGT in a parallel corpus as the PE for the APE triplet.

Considering the previous study, these four edit distance-based noising schemes are combined into a single process without discussing the effectiveness of its respective noising scheme. Regarding this study, we verify the practical effectiveness of each noising scheme by analyzing the performance of the APE models trained by the APE data constructed by each corresponding noising scheme.

3.1.2. POS-based Noising Scheme

The edit distance-based noising scheme has limitations in that the corresponding error type is far weakly related to the practical correction process at the human level. Moreover, it cannot sufficiently reflect the errors that should be revised in the real field. To alleviate such a limitation, we propose human-mimicking noising schemes.

We propose the part-of-speech (POS)-based noising scheme that considers POS tag in injecting the noise. Precisely, we propose POS-based substitution and shifting. This has been proposed to obviate the non-actual errors such as substituting verbs with nouns. Considering this case, by limiting the replaced word to a word having the same POS, it is possible to reduce a case in which a sample that is separated from the actual required proofreading work is generated and to generate a more plausible error.

Regarding the noising process, POS tagging for all TGT in a parallel corpus are proceeded in advance. Therefore, every token in TGT is categorized and accumulated by its POS tag for the construction of the bag of word $L_{pos_k}^{TGT} = \{tgt_i | POS(tgt_i) = pos_k, \forall tgt_i \in TGT\}$ for each respective POS tag pos_k. Based on this, we define POS-based substitution noise (Sub_{POS}) and shifting ($\text{Shift}_{\text{POS}}$) as shown in Eq. (3):

$$N_{L_{pos_k}^{TGT}}(tgt_i) = \begin{cases} tgt \in \bigcup_{TGT \in D} \{L_{POS(tgt_i)}^{TGT}\} & \\ & : \text{Sub}_{\text{POS}} \\ tgt_j \in L_{POS(tgt_i)}^{TGT} & \\ & : \text{Shift}_{\text{POS}} \end{cases} \tag{3}$$

By injecting the noise similarly as Eq. (1), Sub_{POS} replace tgt_i with the random token extracted from the whole training corpus, which has the same POS tag with tgt_i. Considering the $\text{Shift}_{\text{POS}}$, tgt_i is replaced with the random token from the same sentence TGT that has the same POS tag with tgt_i. This is to make the APE model better simulate the work of the actual correction work by selecting the word with the same POS tag when noise is injected through word substitution and shifting.

3.1.3. Semantic Noising Scheme

We also propose semantic noising schemes that substitute tokens into semantically manipulated tokens. These include semantically different and identical tokens with different forms. The main purpose of these noising schemes is to correct a word that has been incorrectly translated into a word with a different meaning or a different tone.

For these, we adopt WordNet (Miller, 1995) information. Specifically, we utilize synonym-, hypernym-, hyponym-, and antonym-based substitution noise, which can be retrieved from the WordNet. Especially, synonym substitution (Sub_{syn}) can deal with formality issues that should select tokens considering the subtle

tone difference. Hypernym substitution ($\text{Sub}_{\text{hyper}}$) can capture an error that occurs by misunderstanding the detailed meaning of the word and is translated with the token in an overly robust meaning. Antonym (Sub_{anto}) and hyponym substitutions (Sub_{hypo}) aim at making semantically different errors. These enable the generated APE data reflecting various errors to be corrected at the human level. The noise injecting process is the same as Eq. (1) and (2). In generating pMT, the substituting tokens are selected from the word list retrieved from the WordNet.

3.2. Training Strategies

Considering our study, we adopted the WMT20 SOTA approach, which is suggested by HW-TSC (Yang et al., 2020), for all of our experiments. We utilized the NMT model as a pre-trained model and fine-tuned the APE task to the corresponding model. Particularly, during the fine-tuning, bottleneck adapter layers (BAL) (Houlsby et al., 2019) were appended to the self-attention structure and feed forward network. All the parameters of the pre-trained NMT model are frozen, and only BAL structures are trained during the fine-tuning process. This can enhance the training efficiency and can attain a higher APE performance than the fine-tuned model (Moon et al., 2021b). During the fine-tuning process, each SRC and pMT is concatenated to make an input sequence. By feeding it, the model is trained to generate PE in a sequence-to-sequence manner (Sutskever et al., 2014).

3.2.1. Static Noise Injection

The most straightforward approach in training the APE model by utilizing noising scheme is generation of data using each noising scheme, followed by the training through it. This indicates a method of generating an APE corpus from a parallel corpus and continuing the training with the same data that constructed in the first phase. We denote this training strategy as static noise injection, as a noise injected in generating pMT and is static over the whole training process. Regarding this case, APE model θ is trained to generate PE in the following process:

$$P_\theta(PE) = \prod_i P(pe_i|SRC, pMT, pe_{<i}, \theta) \quad (4)$$

Equation (4) shows the sequence-to-sequence process of generating PE with SRC and pMT as a given input sequence. However, utilizing such an approach, there exists one major concern about its biasness. Although our noising schemes consider human-level correction, an inherent limitation of using the noising scheme is the probability to be overfitted to specific types of errors. This indicates that only biased correction (considering specific and narrow error types) may be trained to the APE model, rather than the general error correction desired by the noising scheme.

3.2.2. Dynamical Noise Injection

To alleviate the limitations in static noise injection strategy, we propose the dynamic noise injection that successively applies different noising scheme for every epoch. Utilizing this, the APE model θ is trained to generate PE as described in Eq. (5):

$$P_\theta(PE) = \prod_i P(pe_i|SRC, pMT^{(t)}, pe_{<i}, \theta) \quad (5)$$

This shows that different pMT is adopted for every epoch t. Even in utilizing the same noising scheme, the position of the noised and substituted tokens differ because these are determined randomly based on the probability p. Because the noised sentence $pMT^{(t)}$ is different for every epoch t, the APE model can handle various noising schemes and can attain more robust error correction capacity.

4. Experimental Results

4.1. Dataset

We experimented with the Korean-English (Ko-En) pair, where the official APE dataset has not been released. We leveraged the Ko-En parallel corpus distributed by AIHub [2] (Park et al., 2021). This corpus consists of 1.6 M sentence pairs from six different domains, including colloquial, news, dialogue, cultural, ordinance, and official document. Among them, we randomly extracted 18 K, and 2 K data for each domain to construct APE training and test data, respectively. The remaining data were utilized as an NMT training corpus. Regarding the training of both the NMT and APE, 12 K data were randomly extracted from the training corpus for the validation.

Particularly, a commercial system was utilized to generate triplets of the APE test data (SRC, pMT, PE), because there was no officially available APE dataset in Ko-En. Subsequently, we explored the actual performance of the APE model. The data statistics of this data are presented in Table 1.

Dataset		Baseline Performance TER(↓)	BLEU(↑)	# Triplets
NMT Training		-	-	1,482,002
APE Training		-	-	108,000
Test	Google	51.929	33.115	12,000
	Microsoft	59.287	25.130	
	Amazon	59.790	22.192	

Table 1: Data statistics and baseline performance of the Ko-En APE dataset. A low TER indicates a correct translation.

As shown in the test set performance in Table 1, google translator demonstrates the relatively best performance

[2] https://aihub.or.kr/

in both TER and BLEU. Through these datasets, we verify the practical effectiveness of the proposed methods.

4.2. Model and Training Details

NMT Pre-training Regarding the pre-training of the APE model, an NMT model with a vanilla Transformer structure was used (Vaswani et al., 2017). To objectively evaluate the actual performance of the noising scheme-based data generation methodologies proposed in this study, the vanilla transformer model was used as the model baseline. Here, the number of encoder and decoder layers is six, the hidden size is set to 512, and the sentence-piece (Kudo and Richardson, 2018) model which vocab size is set to 50,000 is used to compose the input/output of the model. Considering the NMT training, fairseq (Ott et al., 2019) was used, and one RTX A6000 was used to train within a day, and early stopping was applied based on the validation BLEU score (Papineni et al., 2002).

APE Fine-tuning To fine-tune the APE model, we used the bottleneck adapter layer as in Yang et al. (2020), and the middle size of the layer was set to 64, which was 1/8 of the hidden size of the pre-trained language model (Moon et al., 2021b). Huggingface (Wolf et al., 2019) was adopted to construct the APE model structure, specifically, FSMT model was adopted to build a vanilla transformer structure. The training was conducted using one RTX A6000, and each model was trained within a day by applying early stopping based on the validation BLEU score (Papineni et al., 2002). The final performance evaluation of the model proceeds based on the BLEU and TER (Snover et al., 2006) scores.

4.3. Verification of the Noising Schemes

4.3.1. Inspection of Edit Distance-based Noise

First, we implemented performance verification on four types of edit distance-based noising schemes used in the conventional noise-based APE data generation (Lee et al., 2020). Considering both insertion and deletion affect the length of the sentence, we integrate two noises into the length noise. The experimental results are presented in the first row of Figure 1. In this figure, "All" indicates a conventional edit distance-based noising scheme that combines all noising schemes.

As shown in the experimental results, Sub_{ED} shows the highest performance, which is even higher performance than "All," whereas shifting noise is the most deficient. This shows that utilizing the four noising schemes jointly as in previous studies is not optimal. Applying only a fraction of the noising strategies can further improve performance. These experimental results show that a more detailed discussion on a noising method should be proceeded in generating APE data through a noising scheme to obtain better APE performance.

4.3.2. Noising Scheme Utilizing POS Tagging

We demonstrate that the substitution noise can derive a better APE performance based on the previous experiment, whereas the shifting noise shows the lowest performance. In this experiment, we implement further analyses on the effectiveness of substitution, shifting noise by applying POS tag. The experimental results are shown in the second row in Figure 1.

The results show that Sub_{POS} can derive the best performance, even better than the Sub_{ED}. Considering the noise injection process, we found that Sub_{POS} can effectively maintain the structural consistency of TGT in generating pMT by selecting the replacing words to have the same POS tag. This indicates that noise injection conserving the linguistic structure of TGT is more effective than the conventional noise injection when generating pMT from TGT.

In contrast, Shift_{POS} shows the lowest performance, which is even lower than the Sub_{ED}. This can be interpreted that shifting position of the tokens in a sentence attributed to a significant change in meaning. Shifting tokens with the same POS tag can deteriorate the original meaning by disturbing the original semantic role of each token. This semantical and structural difference leads to a lower performance than the Shift_{ED}, which replaces the order of words without considering the POS. This indicates that the different meaning of the pMT between the PE makes the APE model overly biased toward the pre-trained NMT model. This can also be found when substitution and shifting noises are applied together.

4.3.3. Semantic Noising Utilizing WordNet

Regarding this experiment, we demonstrate the effectiveness of the semantical substitution noise, replacing words by retrieving WordNet. We generate pMT by replacing words in the TGT with the corresponding synonyms, hypernyms, hyponyms, and antonyms. The comparative analyses of APE models trained with each strategy are shown in the third row of Figure 1.

Experimental results imply that a higher APE performance can be obtained by replacing words with maintaining their original meaning. In particular, it can be confirmed that we can attain the best performance by mimicking errors that are corrected at the actual human level, such as Sub_{Syn} or Sub_{Hyp}. Additionally, we found that a considerable performance gap arises between the Sub_{Anto} and Sub_{Syn}. This result shows that semantic impairment should be considered in noise injection-based APE data generation. This indicates that semantic coherence between PE and pMT should be maintained to guarantee the APE performance. This implies that semantics should be considered rather than simply imposing noise arbitrarily in noising scheme-based APE data generation.

4.3.4. Combining Noising schemes

We then inspect whether we can obtain performance enhancement by combining noising schemes. We

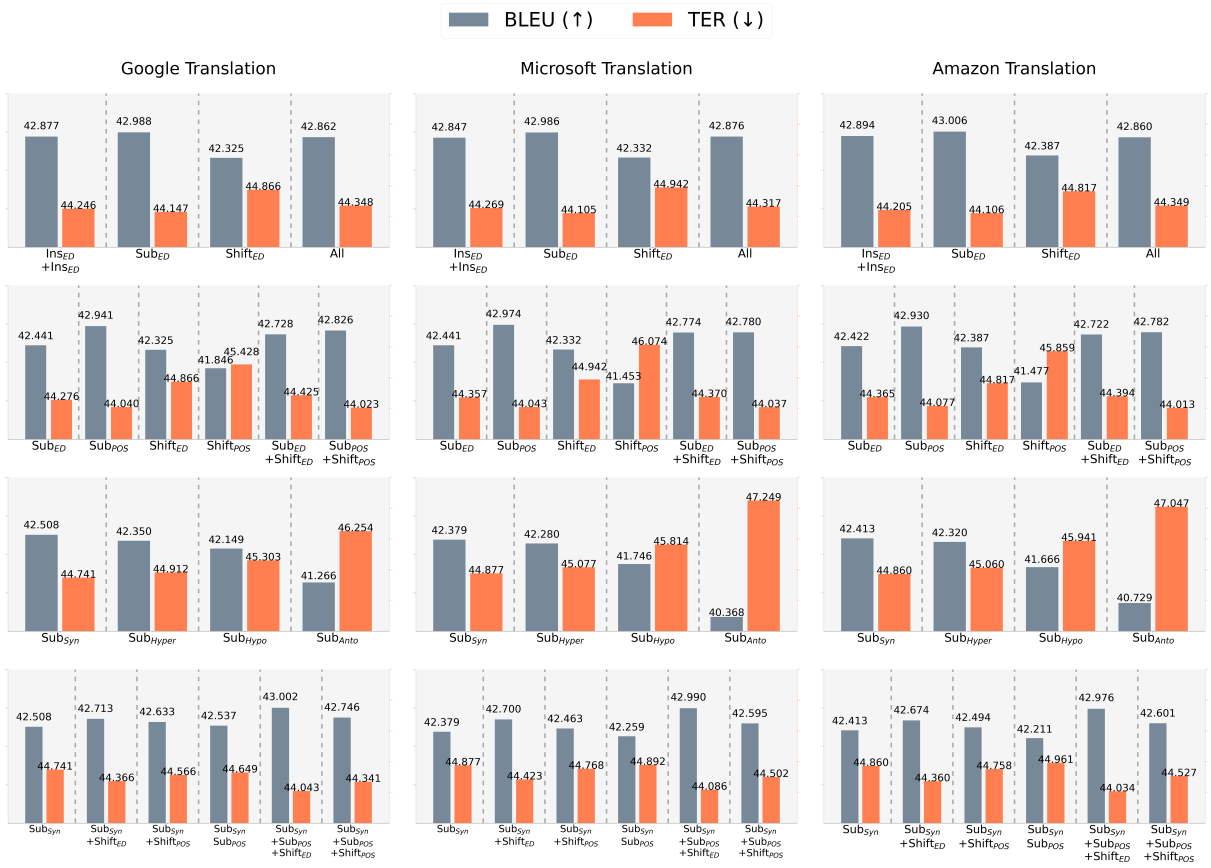

Figure 1: Performance of the APE model trained by each noising scheme. From the leftmost column, each plot shows the post-edited quality of the machine translation results obtained by Google, Microsoft, and Amazon translations.

merged Sub$_{POS}$ with Sub$_{Syn}$ for maintaining semantical and structural coherence in generating pMT. Additionally, we combine shifting noise for validating whether the collaborative effect can be obtained. The experimental results are the same as the last row in Figure 1.

The experimental results indicate that combining shifting noise with substitution noise can yield additional performance improvement. Specifically, combining Sub$_{Syn}$ with Sub$_{POS}$ and Shift$_{ED}$ exhibits the best performance throughout all the experiments. We found that shifting noise provides a positive collaborative effect when it is additionally utilized with substitution noise, while its separate use lead to the low performance. This shows that shifting noise can act positively when semantical and structural coherence is maintained. In this case, APE model can be trained without being overly biased by promoting learning about the correct word order.

4.4. Qualitative Analysis

In addition to the quantitative analysis, we conducted qualitative analysis. The experimental results are as shown in Table 2.

Considering Table 2, the noising scheme that combines Sub$_{\text{POS}}$, Sub$_{\text{Syn}}$, and Shift$_{\text{ED}}$ can yield qualitatively descent quality, compared to PE. This can enable the capturing of the correct sentence structure of PE to obtain the exact meaning and purpose. It shows that precise consideration of the noising scheme can lead to substantially high performance. Moreover Sub$_{\text{POS}}$, Sub$_{\text{Syn}}$, and Shift$_{\text{ED}}$ is the most effective noising scheme that can consider the semantic and structural information.

Particularly, considering the noising schemes that impose semantically promising harms such as Sub$_{\text{anto}}$ or considerably decompose the original sequence structure, such as Shift$_{\text{POS}}$, the quality of the generated sentence is relatively low. Edited sentences for these are far different from the PE and MT sentences. This shows that such noising schemes cannot support model training to correct errors in MT that should be corrected at the human level.

These results suggest that the performance of the APE model differs significantly, depending on the type of noising scheme. The model may be trained in a wrong direction that is out of the original purpose of the APE: correcting errors in MT sentences. This shows that a close inspection of the noising method is required to obtain decent performance when creating an APE

Original Triplet		Example 1	Example 2	Example 3
Original	SRC	더욱 폭넓고 다양한 한국 음악의 세계가 일본에서 펼쳐지길 기대하는 바이다.	현지를 방문해 팬들과 만남을 가지며 자신들의 인기를 실감하게 된 K-POP 스타는 가수 황치열 외에도 '악동뮤지션'이 있다.	국회에서 성토만 할 것이 아니라, 이런 한국 문화 전도사들을 정책적으로 지원할 방도를 찾는 게 옳은 일일 듯싶다.
	PE	I hope that various genres of Korean music will spread in Japan.	The K-POP stars who visited the local area, met with fans, and realized their popularity were "Akdong Musician" and the singer Hwang Chi-yeol.	It might be a good idea to find a way to support these Korean cultural promoters and not just to appeal in the assembly at the National Assembly.
	MT (baseline)	The bar expects a more diverse world of Korean music to be unfolded in Japan.	The K-POP star who visited the local area and met with fans realized their popularity. Besides the singer Huang Qi-yeol, there is "Akdong Musician".	Not just to fill in the National Assembly, it seems the right thing is to find a way to policy-state support for these Korean cultural evangelists.
APE Model		**Example 1**	**Example 2**	**Example 3**
Edited Results	Sub_{ED}	We look forward to seeing a wider and diverse world of Korean music unfold in Japan.	In addition to singer Hwang Chi-yeol, the K-pop star who visited the country met with fans and realized their popularity.	I think it is right to find a way to support these Korean cultural evangelists in a policy manner, not just in the National Assembly.
	Shift_{ED}	We look forward to seeing a wider and more diverse world of Korean music unfold in Japan.	In addition to singer Hwang Chi-yeol, the K-pop star, who visited the local area and met with fans to realize their popularity, has "ak-pop star".	It seems right to find a way to support these Korean cultural evangelists in a policy manner, not just in the National Assembly.
	Sub_{POS}	We look forward to seeing a wider and diverse world of Korean music unfold in Japan.	In addition to singer Hwang Chi-yeol, the K-pop star, who visited the local area and met with fans to realize their popularity, also has "Akdong Musician."	It seems right to find a way to support these Korean cultural evangelists in a policy manner, not just in the National Assembly.
	$\text{Shift}_{\text{POS}}$	The wider and more diverse world of Korean music is expected to unfold in Japan.	In addition to singer Hwang Chi-yeol, the K-pop star, who visited the local area and met with fans to realize their popularity, has "Akdong Musician."	It seems right to find a way to support these Korean cultural evangelists in a policy manner, not just in the National Assembly.
	Sub_{Syn}	The wider and more diverse world of Korean music is expected to unfold in Japan.	In addition to singer Hwang Chi-yeol, the K-pop star who visited the local area met with fans to realize their popularity.	It seems right to find a way to support these Korean cultural evangelists in a policy manner, not just in the National Assembly.
	Sub_{Anto}	The bar and more diverse Korean music world is expected to unfold in Japan.	In addition to singer Hwang Chi-yeol, the K-pop star, who visited the local area and met with fans to realize their popularity, also has "Akdong Music".	Although the National Assembly can be blamed, it seems right to find a way to support these Korean cultural evangelists in a policy manner.
	Sub_{POS} + Sub_{Syn} + Shift_{ED}	It is hoped that a wider and diverse world of Korean music will unfold in Japan.	The K-POP star who visited the local area, met with fans, and realized their popularity, has "Akdong Musician" in addition to singer Hwang Chi-yeol.	It seems that it is right to find a way to support these Korean cultural evangelists in a policy manner, not just a censure in the National Assembly.

Table 2: Qualitative analysis of noising schemes. Baseline indicates the translation results obtained by the Amazon translation system, and the edited results show the corrected sentence of MT generated by the APE model trained through the corresponding noising scheme. Considering these experiments, the best model (blue) and the worst model (red) for each experiment are investigated.

Noising Scheme		**Google**		**Microsoft**		**Amazon**	
		BLEU (↑)	TER (↓)	BLEU (↑)	TER (↓)	BLEU (↑)	TER (↓)
Baseline		33.115	51.929	25.130	59.287	22.192	59.790
Dynamic	Edit Distance Based	42.862	44.348	42.876	44.317	42.860	44.349
	Sub_{POS}	42.941	**44.040**	42.974	**44.043**	42.930	44.077
	Sub_{POS} + Sub_{Syn} + Shift_{ED}	**43.002**	44.043	**42.990**	44.086	**42.976**	**44.034**
Static	Edit Distance Based	42.853	44.153	42.746	44.197	42.870	44.203
	Sub_{POS}	42.874	44.216	42.819	44.265	42.780	44.296
	Sub_{POS} + Sub_{Syn} + Shift_{ED}	42.703	44.430	42.570	44.636	42.556	44.574

Table 3: Performances of the APE models depending on the noise injection strategies.

model based on the noising scheme.

4.5. Effectiveness of the dynamic noising schemes

During the application of these noising schemes to the parallel corpus to train the APE model, we verified the effectiveness of the dynamic training strategy that provided a different noise at every epoch. In addition to the edit distance-based noising scheme used in previous studies, Sub_{POS} and a method combining Sub_{POS}, Sub_{Syn}, and Shift_{ED} was used for the deep inspection. Experimental results are shown in table 3.

As can be seen in our table, the dynamic training strategy that applied a new noise every epoch showed a superior performance than the static strategy. This shows that it is able to learn a wider and more robust error type because semantically and structurally consistent errors are newly applied every epoch. However, regarding the edit distance-based noising scheme, the dynamic strategy showed a lower performance. We interpret that this is because the edit distance-based noising scheme generated pMT by applying noise to TGT without maintaining the structural and semantic consistency. Alternatively, encountering inconsistent type of noises in the training stage actually led to a decrease in the performance. Since inconsistent error types are newly determined every epoch, it seems that the error types that require to be corrected are not properly learned.

5. Discussion

Applying SOTA Approach without Human-edited Data Considering the current APE SOTA approach,

Noising Scheme		Google		Microsoft		Amazon	
		BLEU (↑)	TER (↓)	BLEU (↑)	TER (↓)	BLEU (↑)	TER (↓)
Baseline		33.115	51.929	25.130	59.287	22.192	59.790
Corpus Separating	Edit Distance Based	42.862	44.348	42.876	44.317	42.860	44.349
	Sub_{POS}	42.941	**44.040**	42.974	**44.043**	42.930	44.077
	Sub_{POS} + Sub_{Syn} + $Shift_{ED}$	43.002	44.043	42.990	44.086	42.976	**44.034**
Corpus Overlapping	Edit Distance Based	42.906	44.245	42.909	44.242	42.884	44.256
	Sub_{POS}	**43.011**	44.176	**43.016**	44.190	**43.018**	44.167
	Sub_{POS} + Sub_{Syn} + $Shift_{ED}$	42.882	44.299	42.742	44.493	42.762	44.465

Table 4: Performances of the APE model depending on the corpus separating strategies.

the APE task is fine-tuned to the NMT model for obtaining a better performance (Yang et al., 2020; Oh et al., 2021). Although previous studies had focused on the utilization of multilingual pre-trained language model that trained in a self-supervised learning, as an advent of the NMT-based APE approach (Yang et al., 2020), it was confirmed that leveraging the NMT model could significantly improve the APE performance (Yang et al., 2020; Oh et al., 2021).

However, the NMT-based APE research raises a question in training the APE model without human-edited data as observed in our study. Because a parallel corpus is also utilized in an APE training process, the NMT and APE training corpora overlap. Considering most of existing APE studies where parallel corpus is utilized as an auxiliary corpus for the APE task (Wang et al., 2020; Chatterjee et al., 2019), the usage of parallel corpus is quite clear. However, regarding a situation where a single parallel corpus acts as NMT and APE corpora, more consideration about the training corpus should be made. If we treat the APE training corpus as the same as the NMT training corpus, over-bias problems may occur; nevertheless, there is no experiment about it.

Considering this section, we merge this training corpus selection problem by utilizing the SOTA approach without the human-edited APE data. Throughout the above experiments, we preliminarily separated the NMT and APE corpora prior to the training process. This indicates the splitting of the whole parallel corpus D into D_{APE} and D_{NMT}, where $D_{APE} \cap D_{NMT} = \phi$ and utilizing the respective dataset for the corresponding training. Utilizing these, the training object of the NMT and APE task are shown in Eq. (6) and (7), respectively.

$$\max_{\theta} \sum_{D_{NMT}} \log \left[\prod_{i} P(tgt_i | src, tgt_{<i}, \theta) \right] \quad (6)$$

$$\max_{\theta} \sum_{D_{APE}} \log P_{\theta}(PE) \quad (7)$$

We denote this strategy as the corpus separating strategy. To verify the effectiveness of such a strategy, we trained the NMT and APE models with the same whole training dataset D. Because the training data for these two tasks fully overlapped, we denote it as a corpus overlapping strategy.

Corpus-separating or -overlapping The corresponding experimental results are shown in Table 4. Considering all the training processes, we utilize a dynamic training strategy, which has been shown to be effective in a prior section.

Considering our results, the corpus-separating strategy yields a higher performance for most of the cases, excluding the BLEU score of Sub_{POS}. When the data used for the NMT training are used for the APE learning, it is overly biased to the data, and performance degradation occurs. The above experimental results show that training the NMT and APE models with a smaller amount of data through the corpus-separating strategy can lead to more advantageous results, considering the training time efficiency and performance.

6. Conclusion

We proposed a method to construct the APE data without post-editing the sentences constructed through human labor. Mainly focusing on the noising scheme-based data generation method, we verified the effectiveness of various noising schemes that reflect the human-level error correction process. Through precise inspection, we confirmed that maintaining semantical and structural coherence in imposing noise yields improved performance. We also found that combining Sub_{POS}, Sub_{syn}, and $Shift_{ED}$ can derive optimal performance. In addition, we verified that the dynamic noise injection strategy that injects different noise for each training epoch could achieve a higher performance in the APE model. It has also been shown that separating the NMT and APE corpora is more effective, considering the training time and performance.

However, the performance difference between these various approaches remains relatively small. We also found that the post-edited sentences processed by the APE model become similar for different translation results. We speculate that these results are originated from the unveiled black-box nature of the APE model that fine-tuned to the translation model. We leave it as a future study and plan to figure it out through further analyses.

7. Acknowledgments

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information

Technology Research Center) support program(IITP-2018-0-01405) supervised by the IITP(Institute for Information Communications Technology Planning Evaluation), and the Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(NRF-2021R1A6A1A03045425)

Chatterjee, R., Negri, M., Raphael, R., and Turchi, M. (2018). Findings of the wmt 2018 shared task on automatic post-editing. In *Third Conference on Machine Translation (WMT)*, pages 723–738. Association for Computational Linguistics (ACL).

Chatterjee, R., Federmann, C., Negri, M., and Turchi, M. (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.

Chatterjee, R., Freitag, M., Negri, M., and Turchi, M. (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, November. Association for Computational Linguistics.

Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., Attariyan, M., and Gelly, S. (2019). Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR.

Kudo, T. and Richardson, J. (2018). Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*.

Lee, W., Shin, J., Jung, B., Lee, J., and Lee, J.-H. (2020). Noising scheme for data augmentation in automatic post-editing. In *Proceedings of the Fifth Conference on Machine Translation*, pages 783–788.

Lee, W., Jung, B., Shin, J., and Lee, J.-H. (2021). Adaptation of back-translation to automatic post-editing for synthetic data generation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3685–3691.

Loper, E. and Bird, S. (2002). Nltk: The natural language toolkit. *arXiv preprint cs/0205028*.

Lopes, A. V., Farajian, M. A., Correia, G. M., Trénous, J., and Martins, A. F. (2019). Unbabel's submission to the wmt2019 ape shared task: Bert-based encoder-decoder for automatic post-editing. *arXiv preprint arXiv:1905.13068*.

Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Moon, H., Park, C., Eo, S., Seo, J., Lee, S., and Lim, H. (2021a). A self-supervised automatic post-editing data generation tool. *arXiv preprint arXiv:2111.12284*.

Moon, H., Park, C., Eo, S., Seo, J., and Lim, H. (2021b). An empirical study on automatic post editing for neural machine translation. *IEEE Access*, 9:123754–123763.

Negri, M., Turchi, M., Chatterjee, R., and Bertoldi, N. (2018). Escape: a large-scale synthetic corpus for automatic post-editing. *arXiv preprint arXiv:1803.07274*.

Oh, S., Jang, S., Xu, H., An, S., and Oh, I. (2021). Netmarble ai center's wmt21 automatic post-editing shared task submission. *arXiv preprint arXiv:2109.06515*.

Ott, M., Edunov, S., Baevski, A., Fan, A., Gross, S., Ng, N., Grangier, D., and Auli, M. (2019). fairseq: A fast, extensible toolkit for sequence modeling. *arXiv preprint arXiv:1904.01038*.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (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.

Park, C., Shim, M., Eo, S., Lee, S., Seo, J., Moon, H., and Lim, H. (2021). Empirical analysis of korean public ai hub parallel corpora and in-depth analysis using liwc. *arXiv preprint arXiv:2110.15023*.

Petrov, S., Das, D., and McDonald, R. (2011). A universal part-of-speech tagset. *arXiv preprint arXiv:1104.2086*.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., and Makhoul, J. (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.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Wang, J., Wang, K., Fan, K., Zhang, Y., Lu, J., Ge, X., Shi, Y., and Zhao, Y. (2020). Alibaba's submission for the wmt 2020 ape shared task: Improving automatic post-editing with pre-trained conditional cross-lingual bert. In *Proceedings of the Fifth Conference on Machine Translation*, pages 789–796.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., et al. (2019). Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Yang, H., Wang, M., Wei, D., Shang, H., Guo, J., Li, Z., Lei, L., Qin, Y., Tao, S., Sun, S., et al. (2020). Hw-tsc's participation at wmt 2020 automatic post editing shared task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 797–802.