DbAPE: Denoising-based APE System for Improving English-Myanmar NMT

MAY MYO ZIN, TEERADAJ RACHARAK, and MINH LE NGUYEN
School of Information Science, Japan Advanced Institute of Science and Technology, Ishikawa 923-1292, Japan

Corresponding author: May Myo Zin (maymyozin@jaist.ac.jp).

ABSTRACT Automatic post-editing (APE) research aims to investigate methods for correcting systematic errors in machine translation (MT) results. Recent work has shown successful practices of APE for improving MT output quality; however, their effectiveness strongly relies on the availability of large-scale human-created APE triplets. The high production cost of human post-edited data has led to the absence of APE triplets for most language pairs, including English-Myanmar, which has become a limiting factor for the applicability of the APE task. This work investigates how to conduct the APE task on the English-Myanmar MT where human-edited APE triplets are unavailable. We build an APE system using only the monolingual and parallel MT corpora. The system takes the source sentence (src) and the MT output (mt) as inputs and produces the post-edited mt as output by operating the three processes together, including word alignment extraction, enriching mt using the extracted word alignment information, and denoising the enriched version of mt. We conduct extensive experiments by applying our APE system as a post-processor to the raw output of the existing English-Myanmar MT systems and show that it significantly improves the quality of baseline translation results in terms of TER and BLEU scores. In addition, we perform word alignment experiments with four types of alignment methods and demonstrate that the proposed multilingual word aligner can achieve robust performance over previous state-of-the-art models.

INDEX TERMS Automatic post editing, pre-trained embeddings, bilingual dictionary, word alignment, denoising.

I. INTRODUCTION
Output of machine translation (MT) are plausibly called “pre-translation” as they are not always perfectly correct and might need revisions by human experts for correcting the systematic errors. The goal of APE system is to automatically fix these errors in a machine-translated text by learning from human post-edited samples. Earlier APE researchers adopted the phrase-based statistical machine translation (PBSMT) models to train the APE system as a monolingual re-writing task without considering the source sentence [1], [2]. However, PBSMT-based APE models are only applicable to fix the errors in the output of rule-based MT systems. There are no or only modest improvements while using PBSMT both for first-stage MT and the second stage APE without additional source context modelling and thresholding [3]. The majority of recent APE approaches adopt a dual-source (or multi-source) sequence-to-sequence structure that extends the Transformer [4] in a supervised learning setting [5], [6].

Generally, building an APE system requires a training set comprising the triplets (source-text, MT-output, human post-edit), denoted as (src, mt, pe), respectively. The source sentence (src) and its corresponding MT output (mt) are simultaneously taken as inputs to the APE models and the associated human post-edited sentence (pe) is used as the target. As the high production cost of the target data (pe), the quantity of available APE triplets is still insufficient to train the deep and complex APE models. Currently, strong APE models have failed to show any notable improvement in the refinement of neural machine translation (NMT) output when training on similar-sized human post-edited data [5]-[7].

Open APE triplets are available only for very few language pairs such as English-German and English-Chinese\(^1\). Most of the language pairs including English-Myanmar are absent of APE triplets and thus hinder the applicability of the APE task. To make APE more widely applicable for the most language pairs where APE triplets are

\(^1\)https://statmt.org/wmt21/ape-task.html
unavailable, this work investigates an alternative solution to conduct the APE task without having access to the human-edited APE triplets.

We introduce an easy and effective APE system that uses only monolingual and parallel MT corpus without using any human-edited APE triplets. Our APE system takes the MT output (mt) and its original source sentence (src) as the inputs, and output the high-quality target sentence (post-edited mt) by performing a series of the following three steps:

1. Extracting word alignment information between mt and src using our proposed word aligner,
2. Enriching mt by removing unaligned target words and adding missing source-side information into the target words based on alignment information and bilingual dictionaries for maximizing the semantic similarity between mt and its source sentence, and
3. Denoising the enriched mt (from Step 2) to generate a high-quality target sentence with our proposed denoiser.

Our word aligner is exploited LaBSE [8] which uses cross-lingual word embeddings on a given sentence pair. Regarding the bilingual dictionaries used in the sentence enrichment step, we create two types of bilingual dictionaries from (1) source and target monolingual corpus and (2) parallel MT corpus. For denoisers, we use Transformer [4] models and train them on target monolingual data. The main contributions of this paper are:

- We develop a new word aligner for English-Myanmar using the pre-trained language-agnostic sentence embedding model called LaBSE [8] that leverages effectively to extract the alignment from cross-lingual word embeddings. Our word aligner achieves state-of-the-art performance even in the absence of explicit training on parallel corpus.
- We introduce a simple yet effective method to enrich raw translated text using the bilingual dictionaries extracted from existing monolingual and parallel corpus. Our method effectively considers missing source-side information and context in lexical choices.
- We propose a postprocessor for APE systems that can generate qualified output in the target language using the denoising autoencoder, handling multi-aligned words, and local reordering.
- We verify that cross-lingual embedding on sub-word units performs poorly in word alignment task.
- We empirically show that an APE system built from combining the above three modules is effective and leverages well the existing monolingual corpora, parallel corpus, and pre-trained model; it can be the best learning approach in a low-resource setting where APE triplets are unavailable.

Our proposed APE system can be effectively use as a postprocessor to the raw output of the existing NMT system for most language pairs, without using any human-edited APE triplets. The analyses provided in this work show better understanding of learning the pre-trained model and its usage in the APE task to generate contextualized word embeddings for extracting word alignment information and enriching translated sentences. Moreover, this work demonstrates that the denoising autoencoder, usually used as a language model in various downstream Natural Language Processing (NLP) tasks, can also be applied as a monolingual sentence rewriter in an APE system. Altogether, we show that in a low-resource setting that has only available monolingual and limited parallel data, not only the proposed multilingual word aligner outperforms the existing state-of-the-art models on the word alignment extraction task, but also our denoising-based APE system can help to revise the raw translated texts of existing English-Myanmar MT systems to meet the agreed level quality. As a result of our experiments, this work suggests the optimal research direction in APE for most of the language pairs where human-edited APE triplets are unavailable.

II. MODEL ARCHITECTURE

Our denoising-based APE system (DbAPE) is proposed as a pipeline consisting of three main modules. Fig 1 (a) depicts the first module that performs word alignment information retrieval from an input sentence pair of a source sentence (src) and a machine-translated target sentence (mt), utilizing cross-lingual word embeddings. Fig 1 (b) depicts the second module which removes the typical errors in mt and minimizes the semantic gap between the mt and src by enriching with the missing source-side information. We call this operation target sentence enrichment that enriches mt to be a better version by correcting the errors and adding missing information. Fig 1 (c) depicts the final denoising module where we take the enriched-version sentence (enriched-mt) as input and clean it by removing all possible noises and ordering the words and phrase to be in an acceptable target style.

A. WORD ALIGNMENT INFORMATION RETRIEVAL

Given a pair of source sentence \( x = (x_1, x_2, ..., x_n) \) of length \( n \) and its corresponding parallel target sentence \( y = (y_1, y_2, ..., y_m) \) of length \( m \), the task of word aligner \( A \) is to find a set of pairs of source and target words which are semantically similar to each other within the context of the sentence.

\[
A = \{< x_i, y_j >: x_i \in x, y_j \in y \}
\]

1) EXTRACTING ALIGNMENTS FROM EMBEDDINGS

The pre-trained word embedding models such as BERT [9] and RoBERTa [10] represent words using continuous vectors calculated in context and have achieved impressive performance in a variety of NLP tasks. Multilingually trained sentence embedding models such as language-agnostic BERT, called LaBSE [8], have adapt multilingual BERT (mBERT) [9] to produce language-agnostic cross-lingual...
sentence embeddings for 109 languages, giving the state-of-the-art on the parallel text (bi-text) retrieval task. LaBSE is originally proposed for bi-text process to find the translation pairs in multiple languages. However, this work uses LaBSE for the word alignment extraction task that finds and extracts semantically similar source-target word pairs in a given parallel sentence pair.

Algorithm 1: Word Alignment Information Retrieval

\[
\text{src} = [\text{src}_1, \text{src}_2, ..., \text{src}_n] \\
\text{mt} = [\text{mt}_1, \text{mt}_2, ..., \text{mt}_m] \\
A_{\text{src} \rightarrow \text{mt}} = \emptyset \\
A_{\text{src}} = \emptyset \\
A_{\text{mt}} = \emptyset \\
U_{\text{src}} = \emptyset \\
U_{\text{mt}} = \emptyset \\
\text{for } i = 1 \text{ to } m:\n\text{for } j = 1 \text{ to } n:\n\text{\quad} k = 0 \\
\text{\quad} \text{highest_score} = 0 \\
\text{\quad} \text{score} = \text{cosine_sim}(\text{LaBSE}(\text{mt}_j), \text{LaBSE}(\text{src}_i)) \\
\text{\quad} \text{if score} > \text{highest_score}: \\
\text{\quad} \text{\quad} \text{highest_score} = \text{score} \\
\text{\quad} \text{\quad} k = j \\
\text{\quad} \text{if highest_score} \geq t: \\
\text{\quad} \text{\quad} \text{append } (\text{src}_i, \text{mt}_j) \text{ to } A_{\text{src} \rightarrow \text{mt}} \\
\text{\quad} \text{\quad} \text{append } \text{src}_i \text{ to } A_{\text{src}} \\
\text{\quad} \text{\quad} \text{append } \text{mt}_j \text{ to } A_{\text{mt}} \\
\text{\quad} \text{\quad} \text{for } i = 1 \text{ to } m: \\
\text{\quad} \text{\quad} \text{\quad} \text{if } \text{mt}_j \notin A_{\text{mt}}: \\
\text{\quad} \text{\quad} \text{\quad} \text{append } \text{mt}_j \text{ to } U_{\text{mt}} \\
\text{\quad} \text{\quad} \text{\quad} \text{for } j = 1 \text{ to } n: \\
\text{\quad} \text{\quad} \text{\quad} \text{\quad} \text{\quad} \text{if } \text{src}_i \notin A_{\text{src}}: \\
\text{\quad} \text{\quad} \text{\quad} \text{\quad} \text{\quad} \text{\quad} \text{append } \text{src}_i \text{ to } U_{\text{src}}
\]

While prior works have relied on parallel training data to obtain the word alignments, here we propose a more effective and simpler approach which is particularly suitable for low-resource languages that are lack of the parallel data to train the word aligner. We propose an unsupervised word alignment model that aligns words from the LaBSE based cross-lingual word embeddings. We consider this alignment extraction process as a semantic search task.

In the reminder of the paper, we denote the list of word alignment pairs by \(A_{\text{src} \rightarrow \text{mt}}\) and the lists of aligned source and target words by \(A_{\text{src}}\) and \(A_{\text{mt}}\), respectively. Finally, we denote the list of unaligned source words by \(U_{\text{src}}\) and the list of unaligned target words by \(U_{\text{mt}}\). The detail of our word alignment procedure is described in Algorithm 1, where \(t\) is a user-defined word pair similarity threshold. As cosine similarity score between two word vectors falls in the range of 0 to 1, we set the threshold \(t\) to 0.5, at the halfway mark.

As shown in the algorithm, the word alignment information retrieval task proceeds as follows. Given a pair of source sentence \((\text{src})\) and its corresponding MT output \((\text{mt})\), we extract the most similar \(\text{src}\) word for each \(\text{mt}\) word base on the similarity score computed by the cosine similarity function on their LaBSE based cross-lingual word embeddings. Among the extracted highest similar pairs, the pairs with the similarity score higher than the threshold are considered as the final word-aligned pairs \(A_{\text{src} \rightarrow \text{mt}}\). Meanwhile, we record the unaligned source words \(U_{\text{src}}\) and unaligned target words \(U_{\text{mt}}\) in \(\text{src}\) and \(\text{mt}\), respectively.

B. TARGET SENTENCE ENRICHMENT

This section is to enrich MT output by removing errors and adding missing information based on word alignment information and bilingual dictionaries.

Given the monolingual and parallel corpus, we first build two bilingual dictionaries (cf. Figure 1): a monolingual corpus-based dictionary \((\text{MD})\) and a parallel corpus-based dictionary \((\text{PD})\), by extracting potential source-target translation word pairs with similar vectors.

Using the source and target monolingual texts, we build a bilingual \(\text{MD}\) as follows:

- We create the source and target vocab files which contain the list of source and target words,
- We feed these two files as input into our word aligner (Algorithm 1), and
We store all extracted source-target word alignment pairs in the bilingual MD.

Using the parallel corpus, we build a bilingual PD. From each pair of source sentence \( x \) and target sentence \( y \) in parallel corpus, the potential translated word pairs are extracted as follows:

- We create forward-aligned \( A_{\text{forward}} \) and backward-aligned \( A_{\text{backward}} \) word pairs by running the proposed word alignment information retrieval module (Algorithm 1) in source-to-target and target-to-source directions, respectively, as follows:

\[
\begin{align*}
A_{\text{forward}} &= \{ < x_i, y_j > : x_i \in x, y_j \in y \} \\
A_{\text{backward}} &= \{ < x_i, y_j > : x_i \in y, y_j \in x \}
\end{align*}
\]

- We find the common of all aligned word-pairs \( < x_i, y_j > \) from both lists \( A_{\text{forward}} \) and \( A_{\text{backward}} \) and store them into the bilingual PD, i.e.,

\[
PD = A_{\text{forward}} \cap A_{\text{backward}}
\]

Since bilingual PD is built in the supervised setting with the guidance of the parallel aligned sentence pair, it should be more accurate than MD built in the unsupervised setting and we confirm this hypothesis from our experiments.

**C. TARGET SENTENCE DENOISING**

Although the target sentence enrichment module has removed mistranslated or extra words and added missing source-side information, the enriched-version of MT output (enriched-\( mt \)) is still far from being an acceptable translation. It still needs to improve the word order and perform grammar correction. Moreover, in the appended part of enriched-\( mt \), unaligned source word to similar target word substitution always outputs a target word for every position. There are a plenty of cases that some of the substituted (appended) words should be remove/denoise to make a fluent output. Moreover, in some cases, we need to add extra common words, e.g. prepositions or articles, to be in the correct sentence structure. For example, a sequence of Myanmar source words “နှစ်ယောက်စလို့ဝက်သူတွေကိုကို” would be substituted by word-to-word with the sequence of similar target words “both them to”; however, it must be “both of them” in English. In this case, we consider the substituted word “to” as an insertion noise that needs to remove from the sentence and the extra word “of” as a deletion noise that must be added to the sentence.

To remove the potential noises in enriched-\( mt \), we design a sequence-to-sequence Transformer [4] model that takes a noisy (unstructured) sentence as input and generates a clean (denoised) sentence as output; both of which are of the same (target) language. As shown in Fig 1 (c), we feed the noisy input, enriched-\( mt \), into a designed denoising model so that it transforms the input into a clean target sentence post-edited \( mt \). To inspect the effectiveness of denoising mechanism on improving quality of the final output, we conduct experiments on the denoising task with the following two different models.

1) **DENOISING AUTOENCODER (DA)**

For training the denoising autoencoder, training label sequences would be the target monolingual sentences. Given a clean target sentence, the noisy input should be ideally the unstructured version of the corresponding source sentence. To create the noisy versions, we inject artificial noise into a clean sentence to simulate the noise of our enriched-version sentence.

Firstly, for each sentence in a given monolingual corpus, we remove 20 to 30 percent of out-of-vocabulary (OOV) words and append the deleted words to the end of the sentence. Then, we insert the following artificial noises into the source side:

a) Insertion of random frequent tokens where the model learns to remove extra/redundant words:

1. For each position \( i \), a probability \( p_i \sim \text{Uniform}(0, 1) \) is first sampled,
2. Let \( p_{\text{ins}} \) be a probability threshold of the insertion. If \( p_i < p_{\text{ins}} \), we sample a word \( w \) from the most frequent target words \( V_{\text{ins}} \) and then insert it before the position \( i \).
The inserted words are limited by $V_{ins}$ because target insertion occurs mostly with common words, e.g., prepositions or articles. We threshold the value with $p_{ins}$ to decide for inserting the words.

b) Deletion of tokens helps the model learn to predict and add potential words for fluency:
   1. For each position $i$, a probability $p_i \sim \text{Uniform}(0, 1)$ is first sampled.
   2. Let $p_{del}$ be a probability threshold of the deletion. If $p_i < p_{del}$, we drop the word at the position $i$. We threshold the value with $p_{del}$ to decide for deleting the words.

c) Permutation of tokens with a limited distance is applied to stimulate the learned model to modify the word order in a correct target structure:
   1. Let $d_{per}$ be a degree of the permutation. For each position $i$, an integer $\delta_i \in [0, d_{per}]$ is sampled,
   2. We add $\delta_i$ to index $i$ and sort the incremented indices $i + \delta_i$ in an increasing order.
   3. The words are rearranged in the new positions, to which their original indices have moved by Step 2.

For insertion, deletion, and reordering noises, we adopt the designs and settings of the previous work in [11]. In our target sentence denoising module, we consider a vocabulary size of 32,000 words, and words out of this vocabulary are called OOV words.

2) DENOISING REWRITER (DW)

We design a Transformer-based target-to-target rewriting model and train it to generate the clean target sentence from its noisy-version. For training the rewriting model, we build noisy training data from the target monolingual corpus $D_y$. Firstly, we delete 20 to 30 percent of OOV words from a given sentence $y \in D_y$, and then append the deleted words to the end of $y$. Next, for creating the insertion and deletion noise types, we randomly drop/add some words (up to three words for the sentences with the sentence-length greater than ten). Finally, we swap contiguous words randomly with a probability $p_{swap}$ to introduce some noises to get noisy version $y'$. Note that $p_{swap} = 0.2$ is set. We treat $y'$ as the input and $y$ as the output to train the model. For model inference, we feed the enriched-version of MT output (enriched-mt) into the trained model and generate the clean target sentence, post-edited mt.

III. EXPERIMENT

A. EVALUATION METRIC

For evaluating the performance of our APE system, we use two standard evaluation metrics: BLEU\textsuperscript{2} which measures the degree of n-gram match between the model hypotheses and its target; TER\textsuperscript{3} which measures the number of edits required to change a system output into one of the references. We evaluate the performance of the alignment models using Alignment Error Rate (AER)\textsuperscript{12}.

B. DATASETS

As monolingual data for training our denoisers and creating the bilingual dictionary, we used eight million Myanmar sentences gathered from various sources, including textbooks, Myanmar local news, Myanmar Wikipedia, ALT train data [13], and CC100-Burmese dataset [14]. For the English monolingual corpus, we used ten million sentences which combined ALT train data and randomly extracted sentences from WMT monolingual News Crawl datasets\textsuperscript{4}. We used Moses tokenizer to tokenize English sentences. For Myanmar, we used the UCSYNLP segmenter\textsuperscript{5}.

Parallel data is used to build the baseline NMT systems and the bilingual dictionary, and to train/fine-tune the word aligners. We collected around 224 thousand parallel sentences. Data statistics are shown in Table 1.

| Type   | Data Source          | Total Sentences |
|--------|----------------------|-----------------|
| Train  | Local News and Textbooks | 204,535         |
| Dev    | ALT                  | 1000            |
| Test   | ALT                  | 1017            |

C. MODEL CONFIGURATION

The next subsections provide details about the architecture and training procedure of baseline systems and our models.

1) BASELINE MT SYSTEMS

The performance evaluation of our proposed APE system was conducted based on three different test sets, which were generated by a simple Transformer-based NMT, fine-tuned mT5 and Google Translate translation systems. For training the NMT, we used PyTorch version of the OpenNMT project, an open-source (MIT) neural machine translation framework [15]. The Transformer experiments were run on NVIDIA Tesla P100 GPU with the following settings listed in Table 2. For the mT5 system, we were constrained by computational resources to $mt5$-base, which has 580M parameters. We initialized the pre-trained $mt5$-base model using Hugging Face’s AutoModelForSeq2SeqLM\textsuperscript{6}. We used the AdamW optimizer [30] with a learning rate of 5e−4 and transformer’s get_linear_schedule_with_warmup\textsuperscript{7} scheduler, and fine-tuned the model on 8 epochs with batch size of 16 and 1000 training iterations between checkpoints. Parallel datasets shown in Table 1 are tokenized into sub-word units.

---

\textsuperscript{2} We used an implementation of BLEU from https://github.com/moses-smt/mosesdecoder in our experiments.
\textsuperscript{3} http://www.cs.umd.edu/~snower/tercom/
\textsuperscript{4} http://data.statmt.org/news-crawl/
\textsuperscript{5} http://www.nlpresearch-usc.edu.mm/NLP_UCSY/wsandpos.html
\textsuperscript{6} https://huggingface.co/docs/transformers/main_classes/optimizers/schedule
\textsuperscript{7} transformers.get_linear_schedule_with_warmup
by using SentencePiece\(^6\) and used to train/fine-tune and validate the baseline NMT and mT5 systems.

2) DENOISING MODELS

For denoisers, we used 6-layer Transformer encoder/decoder [4]. Denoising autoencoder\(^9\) is trained using Sockeye [11], [21]. For training the denoising rewriter, we use the same tool and settings as used in the baseline NMT. We used the target-side monolingual data to train the denoising models and treat ALT dev set as the validation data.

3) WORD ALIGNMENT MODELS

We apply the pre-trained LaBSE [8] model to get the cross-lingual word embeddings for word alignment information extraction task. We compared our word alignment model with the following baselines:

1) fast_align [16] is a simple, fast, unsupervised word aligner with reparameterization of IBM Model 2.

2) GIZA++ [17], [18] is an implementation of IBM models. We used five iterations each for Model 1, the HMM model, Model 3, and Model 4 to train GIZA++ by following the previous work of [19].

3) AWE-SoME [20] is a neural word aligner based on multilingual BERT that can extract word alignments from contextualized word embeddings with and without fine-tuning on parallel data.

### IV. EXPERIMENTAL RESULTS

In this section, we first describe the main results of our APE model based on our two different denoising strategies: DA and DW on the output of the three baseline MT systems: NMT, mT5 and Google Translate. Then, we evaluate our alignment model and compare its performance with state-of-the-art works. Additionally, we conduct a series of qualitative analysis and ablation studies on the baseline NMT output to further validate the reliability of our proposed models and to better understand the importance of data preprocessing in the word alignment extraction task.

![Image](https://github.com/yunsukim86/sockeye)

### TABLE II

| Parameters for Training Transformer Models |
|--------------------------------------------|
| - layers 6-8, run size 512, word vec size 512 |
| - transformer_ff 2.048 - heads 8 |
| - encoder_type transformer |
| - decoder_type transformer |
| - position_encoding true - train_steps 200,000 |
| - max_generator_batches 2 - dropout 0.1 |
| - batch_size 4,096 - batch_type tokens |
| - normalization tokens - accum_count 2 |
| - optim adam - adam-beta2 0.998 |
| - decay_method noam - warmup_steps 8,000 |
| - learning_rate 2 - max_grad_norm 0 |
| - param_init 0 - param_init_glorot true |
| - label_smoothing 0.1 - valid_steps 1,000 |
| - save_checkpoint_steps 1,000 |
| - world_size 1 - gpu_rank 0 |

### TABLE III

| Performance of APE Models |
|---------------------------|
| **English→Myanmar** | **Myanmar→English** |
| NMT | TER | BLEU | TER | BLEU |
| 79.713 | 8.11 | 75.238 | 11.59 |
| + APE (with DA) | 65.317 | 13.53 | 58.315 | 16.24 |
| + APE (with DW) | 66.472 | 13.14 | 59.471 | 15.96 |
| mT5 | 65.524 | 13.49 | 60.325 | 15.49 |
| + APE (with DA) | 62.732 | 14.87 | 56.841 | 17.83 |
| + APE (with DW) | 67.141 | 13.41 | 59.816 | 15.81 |
| Google Translate | 73.874 | 9.64 | 62.157 | 15.39 |
| + APE (with DA) | 63.495 | 13.92 | 57.853 | 16.98 |
| + APE (with DW) | 74.281 | 9.51 | 60.174 | 15.74 |

### A. MAIN RESULTS

The overall results of our APE model are reported in Table 3. There are two methods of training the proposed APE system as described in Subsection 2.C: DA and DW. The performance of the models is evaluated with BLEU and TER metrics. Our experiments demonstrate that both versions of APE models improve the quality of the texts generated by the baseline NMT. Our APE model trained with DW showed to give at least +4% BLEU and -16% TER, respectively. When we trained the APE system with DA instead of DW, we could have additional gain around +1% BLEU and -2% TER.

To further validate the effectiveness of our APE systems on the state-of-the-art MT systems, we also conduct APE tasks on the output generated by mT5 and Google Translate. In these cases, our APE system trained with DA can still improve their output quality in both directions. However, APE system with DW fails to improve the quality of mT5 and Google Translate texts in the English-to-Myanmar direction.

Both denoisers are built using the same Transformer architecture but are trained on different noisy datasets. Overall, the insertion/deletion/reordering noise types demonstrate a promising performance while mitigating these noises by using the denoising autoencoder, DA.

### B. WORD ALIGNMENT RESULTS

Multilingual sentence embedding model is a powerful tool that encodes text from different languages into a shared embedding space, enabling it to be applied for a range of downstream NLP tasks, like clustering, text classification, and others, while also leveraging semantic information for language understanding. The existing approaches for generating such embeddings, like MUSE\(^10\) or LASER\(^11\), require parallel data to train for mapping a sentence from one language directly into another language to encourage consistency between the sentence embeddings.

---

\(^6\) https://github.com/google/sentencepiece
\(^9\) https://github.com/yunsukim86/sockeye
\(^10\) https://github.com/facebookresearch/LASER
\(^11\) https://github.com/facebookresearch/MUSE
The pre-trained LaBSE model that leverages recent advances on language model pre-training, using both masked language modeling (MLM) and translation language modeling (TLM) objectives, on a BERT-like architecture and fine-tuned on a translation ranking task, results in a state-of-the-art model that encodes text from different languages into a shared embedding space. In this work, we apply pre-trained LaBSE model to encode both source and target words which have similar meaning, into a shared embedding space.

Given a sentence pair, firstly, we encoded all words in each sentence using LaBSE word embeddings. Then, we extracted all possible parallel source-target word pairs from their embeddings by our designed word aligner. We set the threshold value for word similarity to 0.5, as described in Subsection 2.A. The extracted pairs which had the similarity scores higher than the threshold value were considered as the word-aligned pairs. We evaluated our model by using the AER metric.

Table 4 shows the alignment error rates (AERs) of our models and popular word aligners on ALT test data of the English-Myanmar language pair. The results show that our LaBSE-based word aligner achieves consistent improvements over the state-of-the-art baseline models, demonstrating the effectiveness of our proposed method. The best score is in bold. Surprisingly, the alignments which are directly extracted from LaBSE (i.e., w/o fine-tuning setting) already achieve better performance than the popular statistical word aligner fast_align and GIZA++ without fine-tuning on parallel data. To further investigate the performance in the bilingual setting, we trained/fine-tuned the model using the parallel data shown in Table 1. In bilingual setting, our word aligner achieves the best performance than other models.

| Model     | Setting | AER |
|-----------|---------|-----|
| fast_align | bilingual | 35.4 |
| Giza++    | bilingual | 30.9 |
| AWE-Some  | w/o fine-tuning | 28.2 |
| Ours      | w/o fine-tuning | 26.1 |

C. QUALITATIVE ANALYSIS

Our main results reveal that automatic post editing using the denoising autoencoder (DA) is better than the target-to-target rewrite based denoising model (DW). In this section, we additionally conduct a qualitative analysis to perform a more reliable verification of our proposed framework. We analyzed the actual post editing results of two APE models: DA and DW, which were trained through our created noisy datasets. We present some examples from DA-based and DW-based APE models tested on the output of NMT system in Table 5 and Table 6, respectively. From the tables, TER scores in the mt rows are calculated regarding tgt; boldface words in mt indicate words that need to be corrected to match the human-translated reference sentence, tgt of target-side. We found that the output of the baseline English-Myanmar NMT, mt, undergoes an excessive number of corrections, whereas mt post-edited by our APE models requires fewer corrections. Among these two models, post-editing with DA requires the fewest corrections and can make mt to a more accurate and fluent sentence, which is in turn similar to that of the reference sentence.

D. ABLATION STUDY: DENOISING AUTOENCODER

We tuned each parameter of the noise and combined them incrementally to investigate the effect of each noise type in the denoising autoencoder on the baseline NMT output as shown in Table 7. Firstly, we applied the reordering noise with different values of \( d_{\text{per}} \). A significant improvement was achieved from \( d_{\text{per}} = 5 \) since a local reordering usually involved a sequence of 5 to 6 words. We also tried to train with \( d_{\text{per}} > 5 \) and found that it shuffled too many consecutive words together and thus cannot handle long-range reordering, yielding no further improvement.

Secondly, for the deletion noise, \( p_{\text{del}} = 0.1 \) gave +1.16% BLEU, but it immediately degraded with a larger value; it was hard to observe one-to-many in the similar target word substitution more than once in each sentence pair. Finally, for the insertion noise, we observed the best performance (+1.92% BLEU) with \( V_{\text{ins}} = 10 \). Generally, increasing \( V_{\text{ins}} \) was not helpful since it provided too many variations in the inserted word; it might not be related to its neighboring words.

E. ABLATION STUDY: WORD ALIGNMENTS

In this part, we examined the performance of two different types of pre-trained embedding models, namely mBERT and LaBSE, on the supervised word alignment extraction task with our designed word aligner. mBERT is a transformers model pre-trained on a large multilingual Wikipedia corpus using a masked language modeling (MLM) objective. We used the word embeddings of the 8-layer of mBERT following [20] and 12-layer of LaBSE, respectively. We also examined how the word alignment performance varies with different levels of cross-lingual word embeddings.

As shown in Table 8, we can see that LaBSE can significantly outperforms mBERT by a large margin on English-Myanmar language pairs. Both mBERT and LaBSE can support both English and Myanmar languages in a single model but the embedding vectors spaces of mBERT between languages are not aligned, i.e., the text with the same content in different languages would be mapped to different locations in the vector space. This work shows that LaBSE trained on both monolingual sentences and bilingual sentence pairs using MLM and translation language modeling (TLM) with the primary purpose of parallel sentence retrieval can result in a model that is effective on word alignment extraction even on low-resource languages for which there is no data available during training.
It has been confirmed that eight thoroughbred race horses at Randwick Racecourse in Sydney have been infected with equine influenza.

The cases are the first infections of race horses, despite infecting dozens of recreational horses across NSW and Queensland.

The incident is the first release of dozens of resorts throughout the trying season and is infected with for the first time.

The incident is the first infections of dozens of resorts horses throughout the NSW and Queensland, race horses are infected for the first.

The incident is the first infections of dozens of resorts horses throughout the Queensland, race horses are infected with for the first.

It has been confirmed that eight thoroughbred race horses at Randwick Racecourse in Sydney have been infected with equine influenza.

The cases are the first infections of race horses, despite infecting dozens of recreational horses across NSW and Queensland.

The incident is the first infections of dozens of resorts horses throughout the NSW and Queensland, race horses are infected for the first.

The incident is the first infections of dozens of resorts horses throughout the Queensland, race horses are infected with for the first.
Table VII: APE Results with Different Values of Denoising Parameters for Myanmar-English NMT

| d_per | P_det | V_ins | BLEU  |
|-------|-------|-------|-------|
| 2     |       |       | 12.64 |
| 3     |       |       | 13.02 |
| 5     |       |       | 13.16 |
| 5     | 0.1   |       | 14.32 |
| 5     | 0.3   |       | 13.85 |
| 5     | 0.1   | 10    | 16.24 |
| 5     | 0.1   | 50    | 15.97 |
| 5     | 0.1   | 500   | 15.32 |
| 5     | 0.1   | 5000  | 14.86 |

Table VIII: Alignment Results with Different Word Embeddings

| Embedding Model | Setting | AER1 |
|-----------------|---------|------|
| word level      |         |      |
| mBERT           | w/o fine-tuning | 35.7 |
| LabSE           | bilingual | 26.4 |
| sub-word level  |         |      |
| mBERT           | w/o fine-tuning | 38.6 |
| LabSE           | bilingual | 36.8 |

V. RELATED WORK

Most recent APE studies primarily focus on the techniques to alleviate the data sparsity problems in APE. While recent advances have reported that automatic generation of synthetic APE triplets (src, mt, pe) from parallel corpora based on various noising schemes [22] and addition of synthetic data to genuine data to expand the APE training data [23]-[25] can mitigate the data scarcity, other studies have highlighted several open challenges [26]. A major challenge is the quality of the generated synthetic data. These recent synthetic data generation works neglect to comply with minimum-editing criterion, where pe should be created by minimally editing mt yet maintaining the meaning of src. Therefore, the correction patterns detected in this synthetic data may differ from those occurring in the genuine APE data, and possibly limit the APE performance. Moreover, in the case of generating the APE triplets using the existing parallel data, training baseline MT and APE models on the same data size will not be effective [5]-[7]. There is also an issue that pe should not be a reference translation (target text translated by human) in the APE task, since this would defeat the purpose of learning editing patterns for the MT output [27]. In this work, considering the limitations in APE triplet generation and avoiding the absence of APE triplets that hinder the applicability of the APE task on English-Myanmar NMT, we pursue an alternative solution to design an APE model using only available monolingual and parallel data but without using any human-edited APE triplets.

Primarily, APE systems are employed for improving MT output by exploiting information unavailable to the decoder and coping with systematic errors including adequacy and fluency errors of an MT system whose decoding process is not accessible. For this purpose, the previous work on English-French APE [3] tried to maintain the connection between MT output and the source sentence using word alignment information in order to improve the adequacy. They created a new intermediate sentence by concatenating each word in MT output with “#” and aligned source word. Then, their APE model is trained to rewrite the intermediate sentence to reference target sentence. However, their APE pipelines failed to improve on the MT baseline for the English-to-French direction and achieved only a small increase in BLEU of 0.65 absolute over its baseline for French-to-English direction. Parton et al. [28] also tackled specific linguistic adequacy errors. They tried to correct the errors by either replacing or inserting words into the hypothesis. This system only fixed certain word-choice errors (e.g. numbers, names and named entities) using the three resources such as the phrase table, dictionaries and background MT corpus. In our work, we focus on the same purpose but consider an alternative approach to be useful even in the low-resource setting where APE triplets are unavailable. We design a simple and effective word aligner for extracting word alignment information between the MT output and its original source sentence. Using word alignment information, we can perform deeper text analysis. All possible systematic errors such as extra information (unaligned target words) and missing source information (unaligned source words) in the MT output can be examined from the word alignment information. Based on this analysis, we design a sentence enrichment module that enables to enrich the MT output by removing the errors and adding missing information. Moreover, not only for solving the noises but also for transforming enriched MT output into a more accurate and fluent style, we also propose the denoisers to clean the errors and reordering noises. We show that APE systems can adapt the output of a general-purpose MT system to the lexicon/style requested in a specific application domain.
A large body of literature has studied using pre-trained contextualized word embeddings derived from multilingually trained language models (LM) for extracting word alignment information. In the field of neural word alignment, Sabet et al. [29] proposed methods to align words using multilingual contextualized embeddings and achieved competitive results even in the absence of explicit training on parallel data. Recently, Dou et al. [20] proposed a neural word aligner that leveraged pre-trained mBERT and fine-tuned embeddings on the parallel corpus for better alignment results. Although mBERT has shown a reasonable capability for the zero-shot cross-lingual transfer when fine-tuned on the downstream NLP tasks, it is not pre-trained with explicit cross-lingual supervision, and thus transferred performance can further be improved by aligning mBERT with cross-lingual signal. Instead of mBERT, we use LaBSE embeddings in our word alignment extraction task. LaBSE is a powerful model that encodes text from different languages into a shared embedding space and is a new state of the art on the multiple parallel sentence pair retrieval task. It is effective even on the low-resource languages for which there is no data available during training. The experimental results show that LaBSE word embeddings is superior to mBERT in our proposed word alignment extraction task.

The word-by-word translation output of an unsupervised MT system trained only on monolingual corpora can be improved with the denoising autoencoder (Kim et al., 2019). Denoising autoencoder is a sequence-to-sequence neural network model that takes a noisy sentence as input and produces a clean sentence as output, both of which are of the same language. In our APE system, the target sentence enrichment module enriches the raw MT output by deleting unaligned target words and appending the most similar target word for each unaligned source word. This sentence enrichment task of unaligned source words to the closest target words substitution can be considered as the part of word-by-word translation. Following the same idea, we use the denoising autoencoder in our APE system to transform the enriched-version of MT output into a clean and fluent version. As an alternative to the denoising autoencoder, we further design a denoising rewrite, a target-to-target rewriting model and train with different settings of noises. As a result of our experiments, our proposed APE system with the denoising autoencoder (DA) can improve the quality of the texts generated by the state-of-the-art MT systems.

VI. CONCLUSION

In this paper, we propose a simple yet effective APE pipeline that can correct the errors in the translation results of current English-Myanmar NMT systems and greatly improve the quality of their translated texts in both directions. We identify three principles (namely, word alignment, sentence enrichment, and sentence denoising) underlying recent successes in the absence of APE triplets and show how to apply them to build an APE system without having the triplets. In essence, we firstly introduce a neural word aligner that extracts alignments’ information from LaBSE-based contextualized cross-lingual word embeddings. Using the extracted word alignments’ information, we analyze the gap between MT output and its corresponding source sentence. From the analysis, we thereby design the target sentence enrichment module that improves the raw MT text by further removing extra information and inserting missing information. Finally, the enriched-version of MT text is denoised by our proposed denoisers. The final output of our denoiser is a clean and fluent target sentence. Ablation studies show that our APE model integrated with the three principles gives a promising performance even in the absence of APE triplets.

To the best of our knowledge, this is the first attempt of adapting contextualized cross-lingual word embeddings and denoising mechanisms for the APE task on low-resource language pairs like English-Myanmar. We believe that our findings can encourage further research along this direction. The proposed word aligner and denoisers can be effectively applied not only in the APE task but also in other MT-related works. These models can easily be trained in both low and rich-resource settings.

ACKNOWLEDGMENT

The authors would like to thank the Ministry of Education, Culture, Sports, Science and Technology (MEXT) of Japan for providing the Japanese Government (Monbukagakusho) Scholarship under which this work was carried out.

REFERENCES

[1] M. Simard, C. Goutte, and P. Isabelle, “Statistical Phrase-based Post-editing,” in Proc. NAACL HLT 2007:508–515, 2007.
[2] M. Smard, N. Ueffing, P. Isabelle, and R. Kuhn, “Rule-based translation with statistical phrase-based Post-editing.” In: Proceedings of the second workshop on statistical machine translation, Prague, Czech Republic, 2007, pp 203–206.
[3] H. Béchara, Y. Ma, and J. van Genabith, “Statistical post-editing for a statistical MT system,” In: Proceedings of the 13th machine translation summit (MT Summit XIII), Xiamen, China, 2011, pp 308–315.
[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” In Advances in Neural Information Processing Systems 30 (NIPS 2017), pages 5998–6008, Long Beach, CA, USA, 2017.
[5] R. Chatterjee, M. Negri, R. Rubino, and M. Turchi, “Findings of the WMT 2018 shared task on automatic post-editing,” in Proc. 3rd Conf. Mach.Transl., Shared Task Papers, 2018, pp. 723–738.
[6] R. Chatterjee, C. Federmann, M. Negri, and M. Turchi, “Findings of the WMT 2019 shared task on automatic post-editing,” in Proc. 4th Conf. Mach. Transl. (Shared Task Papers, Day), vol. 3, 2019, pp. 11–28.
[7] J. Iye, L. Specia, S. Szoc, T. Vanallemeersch, J. V. den Bogaert, E. Farah, C. Maroti, A. Ventura, and M. Khalilov, “A post-editing dataset in the legal domain: Do we underestimate neural machine translation quality?” In Proceedings of the 12th Language Resources and Evaluation Conference, 2020.
[8] F. Peng, Y. Yang, D. Cer, N. Arivazhagan, and W. Wang, “Language-agnostic bert sentence embedding,” arXiv preprint arXiv:2007.01852, Jul 3, 2020.
[9] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics, 2019.

[10] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “RoBERTa: A robustly optimized BERT pretraining approach,” arXiv preprint, 2019.

[11] Y. Kim, J. Geng, and H. Ney, “Improving Unsupervised Word-by-Word Translation Using Language Model and Denoising Autocoding,” EMNLP 2018.

[12] D. Vilar, M. Popović, and H. Ney, “AER: Do we need to improve our alignments?”, In Proceedings of the International Workshop on Spoken Language Translation, 2006.

[13] H. Riza, M. Purwoadi, T. Uliniansyah, A. A. Ti, S. M. Aljunied, L. C. Mäi, V. T. Thang, N. P. Thai, V. Chea, S. Sam, and S. Seng, “Introduction of the Asian language treebank,” In 2016 Conference of The Oriental Chapter of International Committee for Coordination and Standardization of Speech Databases and Assessment Techniques (O-COCOSDA) 2016 Oct 26 (pp. 1-6). IEEE.

[14] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, “Unsupervised cross-lingual representation learning at scale,” arXiv preprint arXiv:1911.02116. Nov 5, 2019.

[15] G. Klein, Y. Kim, Y. Deng, V. Nguyen, J. Senellart, and A. M. Rush, “OpenNMT: Neural machine translation toolkit,” arXiv preprint arXiv:1805.11462. May 28, 2018.

[16] C. Dyer, V. Chahuneau, and N. A. Smith, “A simple, fast, and effective reparameterization of IBM model 2,” In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics, 2013.

[17] F. J. Och and H. Ney, “A systematic comparison of various statistical alignment models,” Computational Linguistics 29, no. 1, pp.19-51, Mar. 2003.

[18] Q. Gao and S. Vogel, “Parallel implementations of word alignment tool,” In Software Engineering, Testing, and Quality Assurance for Natural Language Processing, pp. 49-57, Jun. 2008.

[19] T. Zenkel, J. Wuebker, and J. DeNero, “End-to-end neural word alignment outperforms GIZA++,” arXiv preprint arXiv:2004.14675. Apr 30, 2020.

[20] Z. Y. Dou, and G. Neubig, “Word alignment by fine-tuning embeddings on parallel corpora,” arXiv preprint arXiv:2101.08231. Jan 20, 2021.

[21] F. Hieber, T. Domhan, M. Denkowski, D. Vilar, A. Sokolov, A. Clifton, and M. Post, “Sockeye: A toolkit for neural machine translation,” arXiv preprint arXiv:1712.05690. 2017 Dec 15.

[22] H. Moon, C. Park, S. Eo, J. Seo, and H. Lim, “An Empirical Study on Automatic Post Editing for Neural Machine Translation,” IEEE Access. 2021 Sep 3:9:123754-63.

[23] M. Junczys-Dowmunt and R. Grundkiewicz, “Log-linear combinations of monolingual and bilingual neural machine translation models for automatic post-editing,” in Proc. 1st Conf. Mach. Transl., Shared Task Papers, vol. 2, 2016, pp. 751–758.

[24] M. Negri, M. Turchi, R. Chatterjee, and N. Bertoldi, “ESCAPE: A large scale synthetic corpus for automatic post-editing,” in Proc. 11th Int. Conf. Lang. Resour. Eval., 2018, pp. 24–30.

[25] W. Lee, J. Shin, B. Jung, J. Lee, and J.-H. Lee, “Noising scheme for data augmentation in automatic post-editing,” in Proc. 5th Conf. Mach. Transl., 2020, pp. 783–788.

[26] W. Lee, B. Jung, J. Shin, and J. H. Lee, “RESHAPE: Reverse-Edited Synthetic Hypotheses for Automatic Post-Editing,” IEEE Access. 2022 Feb 25:10:28274-82.

[27] F. do Carmo, D. Shterinov, J. Moorken, J. Wagner, M. Hossari, E. Paquin, D. Schmidtke, D. Groves, and A. Way, “A review of the state-of-the-art in automatic post-editing”, Machine Translation, 35(2), pp.101-143, Jun. 2021.

[28] K. Parth, N. Habash, K. Mckeown, and G. Iglesias G, “Can automatic post-editing make MT more meaningful?,” In: Proceedings of the 16th annual conference of the European Association for Machine Translation (EAMT 2012) pp 111–118.

[29] M. J. Sabet, P. Dufter, F. Yvon, and H. Schütze, “SimAlign: High quality word alignments without parallel training data using static and contextualized embeddings,” arXiv preprint arXiv:2004.08728. Apr 18, 2020.

[30] I. Loshchilov and F. Hutter, “Fixing weight decay regularization in adam,” CoRR, abs/1711.05101, 2017.

MAY MYO ZIN received the B.C.Sc., B.C.Sc.(Hons.) and M.C.Sc. degrees in computer science from University of Computer Studies, Maubin, Myanmar, in 2009, 2010 and 2012, respectively. She is currently pursuing the Ph.D. degree with the Graduate school of Information Science, JAIST, Ishikawa, Japan. Her research interests include machine learning, data mining, natural language processing, machine translation, and artificial intelligence.

TEERADAJ RACHARAK received his B.Eng. in Software and Knowledge Engineering from Kasetsart University, his M.Eng. in Computer Science from Asian Institute of Technology, his first Ph.D. in Information Science from JAIST in 2018 and the second Ph.D. in Engineering and Technology from Thammasat University in 2019. He is currently working as an Assistant Professor with the Graduate School of Information Science, JAIST and also as an Adjunct Faculty with the Information and Communication Technologies Department, Asian Institute of Technology. His research interest is centered in artificial intelligence (AI) including machine learning, argumentation, description logic, neuro-symbolic AI, and explainable AI.

MINH LE NGUYEN, received the BSc in Computer Science from Hanoi National University, Hanoi, Vietnam, in 1998, Master degree from College of Technology, Vietnam National University, Hanoi, Vietnam, in 2001, and Ph.D. degree in information science from Graduate School of Information Science, JAIST, Ishikawa, Japan, in 2004. He is currently working as a Professor with the Graduate School of Information Science, JAIST, Ishikawa, Japan. His research interests include machine learning, text summarization, machine translation, natural language understanding, artificial intelligence, legal engineering, and grammatical analysis of music.