Modularized Multilingual NMT with Fine-grained Interlingua

Sungjun Lim Yoonjung Choi Sangha Kim
Samsung Research, Seoul, Republic of Korea
{sungjunj.lim, yj0807.choi, sangha01.kim}@samsung.com

Abstract

Recently, one popular alternative in Multilingual NMT (MNMT) is modularized MNMT that has both language-specific encoders and decoders. However, due to the absence of layer-sharing, the modularized MNMT failed to produce satisfactory language-independent (Interlingua) features, leading to performance degradation in zero-shot translation. To address this issue, a solution was proposed to share the top of language-specific encoder layers, enabling the successful generation of interlingua features. Nonetheless, it should be noted that this sharing structure does not guarantee the explicit propagation of language-specific features to their respective language-specific decoders. Consequently, to overcome this challenge, we present our modularized MNMT approach, where a modularized encoder is divided into three distinct encoder modules based on different sharing criteria: (i) source language-specific (Enc_s); (ii) universal (Enc_{all}); (iii) target language-specific (Enc_t). By employing these sharing strategies, Enc_{all} propagates the interlingua features, after which Enc_t propagates the target language-specific features to the language-specific decoders. Additionally, we suggest the Denoising Bi-path Autoencoder (DBAE) to fortify the Denoising Autoencoder (DAE) by leveraging Enc_t. For experimental purposes, our training corpus comprises both *En*-to-*Any* and *Any*-to-*En* directions. We adjust the size of our corpus to simulate both balanced and unbalanced settings. Our method demonstrates an improved average BLEU score by "+2.90" in *En*-to-*Any* directions and by "+3.06" in zero-shot compared to other MNMT baselines.

1 Introduction

Neural Machine Translation (NMT) has been a cornerstone of machine translation success (Bahdanau et al., 2014; Luong et al., 2015b; Vaswani et al., 2017). The success of NMT systems lies in their end-to-end encoder-decoder architecture (Kalchbrenner and Blunsom, 2013; Sutskever et al., 2014; Cho et al., 2014). Furthermore, Multilingual Neural Machine Translation (MNMT) has gained significant attention due to its ability to perform multitasking by translating multiples language pairs using a single model (Dabre et al., 2020).

Johnson et al. (2017) propose *Encoder-Decoder Full Layer Sharing* where a single encoder-decoder translates multiple language pairs. They also prefix the input sentence tokens with a target language token and perform *zero-shot* translation, where the model trains with the source and target languages paired to a pivot language rather than directly paired with each other.

Modularized MNMT (Lyu et al., 2020; Firat et al., 2016a) utilizes language-specific encoders and decoders to alleviate the capacity constraints in MNMT. By doing so, it demonstrates its potential to address the capacity limitations in the MNMT domain.

Nevertheless, it possesses certain drawbacks such as poor zero-shot performance and increased training complexity (Dabre et al., 2020; Qu and Watanabe, 2022). Liao et al. (2021) argued that the lack of layer sharing hinders encoders from producing language-independent (*interlingua*) features, leading them to develop a modularized encoder consisting of two distinct encoder modules based on sharing strategies: (i) source language-specific (Enc_s); (ii) universal (Enc_{all}), as illustrated in figure 1 (a). To enhance the zero-shot performance for English-centric directions, they also introduced the Denoising Autoencoder (DAE), as illustrated in figure 2 (a).

However, according to Zhu et al. (2020), the interlingua feature generated by Enc_{all} might lose the language signal, which can potentially lead to overall possible performance deterioration. Moreover, the DAE of the complete sharing of the upper encoder layers scarcely produces language-specific

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 5884–5899

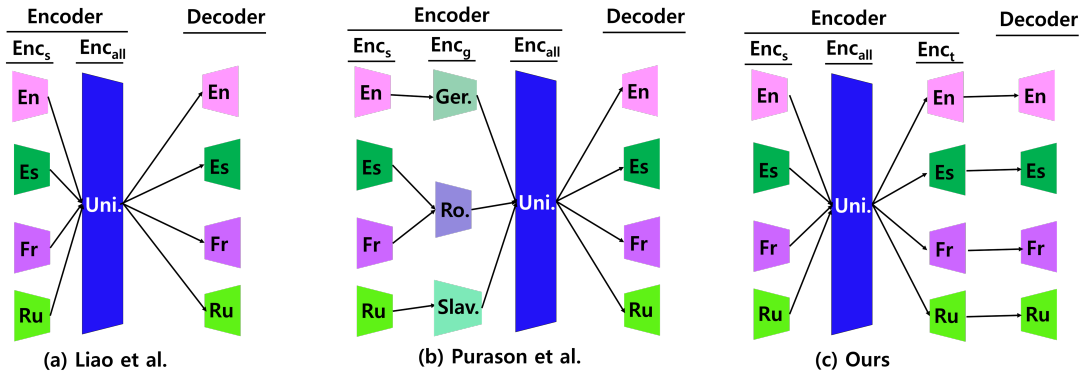

Figure 1: Illustration of the modularized encoders divided into multiple encoder modules by the sharing strategies. (a) Enc_s: source language-specific sharing, Enc_{all}: universal sharing (Liao et al. (2021)) (b) Enc_s: source language-specific sharing, Enc_g: source language group sharing, Enc_{all}: universal sharing (Purason and Tättar (2022)) (c) Enc_s: source language-specific sharing, Enc_{all}: universal sharing, Enc_t: target language-specific sharing (Ours)

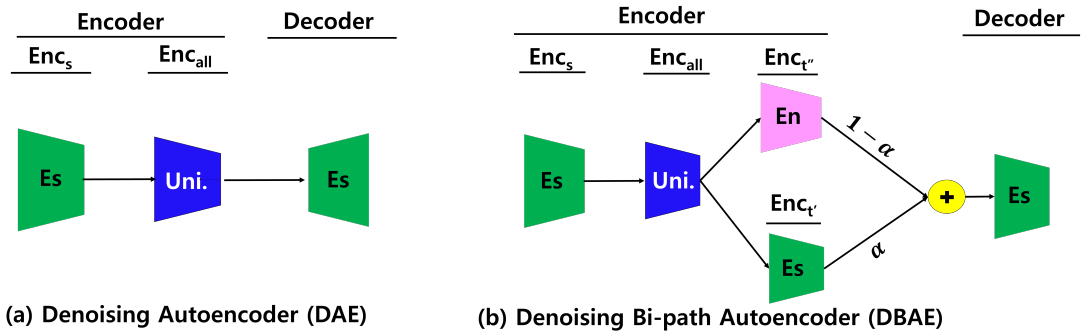

Figure 2: Illustration of different Denoising Autoencoders. (a) Denoising Autoencoder (DAE) with Enc_s and Enc_{all}. (b) Our Denoising Bi-path Autoencoder (DBAE) where output of Enc_{all} bifurcates toward both $Enc_{t''}$ (e.g., English in our case) and $Enc_{t'}$. Outputs from both $Enc_{t''}$ and $Enc_{t'}$ are weighted-summed before the decoder.

features, making it difficult to align with the features that the non-pivot language-specific decoders prefer to accept.

In this work, to alleviate these shortcomings, we propose a new modularized encoder that is divided into three encoder modules with different sharing strategies as follows: (i) source language-specific (Enc_s); (ii) universal (Enc_{all}); and (iii) target language-specific (Enc_t). Our proposed modularized encoder propagates interlingua features through Enc_{all}, and the target language-specific language features to the language-specific decoders through Enc_t.

Besides, to resolve the alignment issue from the DAE, we propose a Denoising Bi-path Autoencoder (DBAE) as illustrated in Figure 2 (b). In the DBAE, the output of Enc_{all} bifurcates and is propagated to both the target language-specific ($Enc_{t'}$) and the pivot language-specific ($Enc_{t''}$) encoder modules. The outputs of both $Enc_{t'}$ and $Enc_{t''}$ are weighted-summed, and the decoder utilizes the weighted-summed features.

By incorporating the $Enc_{t''}$, we attempt to align the encoder output feature with the pivot language-specific feature that the non-pivot language-specific decoders prefer. We provide further details regarding the implementation of DBAE in 3.2.

In this work, we use an English-centric dataset to train all models for experiments. To assess our method, we make use of other MNMT methods as baselines for the experiment, which include (i) Johnson et al. (2017), (ii) Lyu et al. (2020), (iii) Purason and Tättar (2022), and (iv) Liao et al. (2021). Through the experiments, we demonstrate that our modularized MNMT outperforms other MNMT baselines across both English-centric and zero-shot test sets.

2 Related Work

2.1 Modularized Multilingual Neural Machine Translation

The modularized MNMT uses multiple language-specific encoders and decoders (Firat et al., 2016a; Lyu et al., 2020). For example, in the initial phase of the modularized MNMT, Luong et al. (2015a) utilized multiple encoders and decoders for single language pair translation while simultaneously performing multitasking with Autoencoder (AE).

Firat et al. (2016a) introduced the Modularized MNMT, which was realized via single cross-lingual attention shared among all language pairs. Lyu et al. (2020) revisited the modularized MNMT and investigated interlingual representations by exploiting all possible language directions. Furthermore, Lyu et al. (2020) concluded that the modularized MNMT is less vulnerable to the *Capacity Bottleneck*.

2.2 Zero-Shot Translation

Although the modularized MNMT exhibited promising improvement through supervised learning, it did not inherently ensure enhancements in zero-shot performance (Liao et al., 2021). In their work, Firat et al. (2016b) introduced the *Early Average* scheme, wherein a decoder leverages context vectors obtained by averaging both the source and pivot language inputs.

They discovered that the *Early Average* scheme led to enhanced zero-shot performance. Nonetheless, the *Early Average* approach necessitated additional computations to translate the source language input into the pivot language. Our DBAE tackles the problem of extra computations by considering only the source language input. We delve deeper into our DBAE in section 3.2.

The interlingua features are crucial for good zero-shot performance. For example, under the same model architecture as Firat et al. (2016a), Lu et al. (2018); Vázquez et al. (2018) explicitly implemented a single *interlingua* module that converts embedding outputs to language-agnostic features immediately before decoders. Simultaneously, Lu et al. (2018); Vázquez et al. (2018) also utilized an autoencoder to preserve the language-agnostic feature for zero-shot performance.

Instead of implementing an additional module, Liao et al. (2021) suggested a modularized encoder in which the layers before the decoder are universally shared (Enc_{all}). They also multitasked both the machine translation and the DAE to enhance the zero-shot performance.

As an extension of Liao et al. (2021), Purason and Tättar (2022) developed the modularized encoder where middle layers (Enc_g) are shared based on the source language group, as shown in Figure 1 (b). Since they did not use English-centric data, they did not utilize any DAE or AE. They stated that their modularized encoder ensured zero-shot performance when training the model with all possible translation directions.

2.3 Language-Specific Module

Language-Specific (LS) module inserts extra language-specific modules between each layer of both the encoder and decoder (Bapna et al., 2019; Philip et al., 2020; Jin and Xiong, 2022; Zhang et al., 2021; Üstün et al., 2021) or between the encoder and decoder modules (Zhang et al., 2020; Blackwood et al., 2018). Furthermore, Qu and Watanabe (2022) implemented the LS modules within the decoder.

For the modularized MNMT, Purason and Tättar (2022) developed the modularized encoder where the middle of the encoders is shared among source language groups. Instead of layer sharing, Yuan et al. (2023) introduced a detachable model comprising language-specific encoders/decoders and single multilingual encoder/decoder. Nonetheless, they did not address the zero-shot translation. Our modularized encoder addresses the zero-shot translation by the English-centric directions using the DBAE that utilizes two Enc_t modules.

3 Method

This section introduces how to implement our modularized MNMT. We provide details about our modularized encoder and then explain our Denoising Bi-path Autoencoder (DBAE). Lastly, we put forth different inference hypotheses for zero-shot.

3.1 Universal Sharing in Middle of Encoder

Given N languages such that $s, t \in \{l_1, l_2 .., l_N\}$ where s, t standing for source and target language of the parallel corpus, Enc_s is the encoder layers shared according to source language s, and Dec_t is the decoder layers shared according to target language t. Enc_{all} is the encoder layers shared by all languages. As illustrated in Figure 1 (c), our modularized encoder consists of multiple encoder modules as follows: (i) Enc_s; (ii) Enc_{all}; (iii)

Enc_t.

Given M is the number of all possible English-centric language pairs, let $D = \{ec_1, ec_2 .., ec_M\}$ where each ec_i denotes the English-centric pair. Given corpus C_{ec_i} of the English-centric language pairs ec_i from D, our modularized MNMT is trained by maximizing likelihood L_{mnmt}:

$$L_{mnmt} = \sum_{x,y \in C_{ec_i}, ec_i \in D} \log(p(y|x;\theta)), \quad (1)$$

where

$$p(y|x) = softmax(f(x)) \quad (2)$$

, given

$$f(x) = Dec_t(Enc_t(Enc_{all}(Enc_s(x)))) \quad (3)$$

During the training, each training batch only contains single language pair s, t. Then, Enc_s, Enc_{all}, and Enc_t are selected to compose the modularized encoder, after which the modularized encoder propagates output to the Dec_t. In Equation (3) and Figure 1 (c), we can see that our modularized encoder has the encoder modules shared differently compared to other modularized encoders (Liao et al., 2021; Purason and Tättar, 2022).

Enc_t propagate the target language-specific features to the language-specific decoders. Therefore, similar to what Zhu et al. (2020) indicated, we expect the improvement by Enc_t. In the following experiment, we verify whether Enc_t improves the performance compared to the baselines.

3.2 Denoising Bi-path Autoencoder

Liao et al. (2021); Purason and Tättar (2022) mentioned that the DAE is necessary for the zero-shot on English-centric data. In this work, we propose Denoising Bi-path Autoencoder (DBAE) to improve the zero-shot performance. To train the DBAE, a monolingual dataset of N languages is first collected. To noise input sentence, we first tokenize the input sentence and apply the same noise function as Liao et al. (2021).

Inspired by the *Early Average* (Firat et al., 2016b) wherein two distinct context vectors are weighted-summed, the DBAE integrates a bi-path where the output from the Enc_{all} bifurcates to the target language-specific encoder module ($Enc_{t'}$) and pivot language-specific encoder module ($Enc_{t''}$). Thus, outputs from two encoder modules are weighted-summed (e.g., by α for $Enc_{t'}$ and $1-\alpha$ for $Enc_{t''}$) to form a single-path.

The distinction between the *Early Average* and the DBAE lies in the fact that the DBAE uses single input. Consequently, unlike the *Early Average*, our DBAE does not require the additional computation mentioned in section 2.2. Assume x_{noise} is a monolingual sentence corrupted by the noise. And, L is a set of languages for the training. Given monolingual corpus C_i of language i from L, our DBAE is trained by maximizing the likelihood L_{dbae}:

$$L_{dbae} = \sum_{x \in C_i, i \in D} \log(p(x|x_{noise};\theta)), \quad (4)$$

where

$$p(x|x_{noise}) = softmax(f(x_{noise})), \quad (5)$$

given

$$f(x) = Dec_t(\alpha * t(x) + (1-\alpha) * v(x))$$
$$t(x) = Enc_{t'}(Enc_{all}(Enc_s(x)))$$
$$v(x) = Enc_{t''}(Enc_{all}(Enc_s(x)))$$

In every DBAE training, the input language is uniformly randomly chosen out of N languages. To train both L_{mnmt} and L_{dbae}, we adopt the alternating training method proposed by (Dong et al., 2015) . With the help of the DBAE, we anticipate achieving alignment between the output of our encoder and the pivot language-specific features favored by non-pivot language-specific decoders. In section 5.3, we discuss whether the DBAE can improve the zero-shot performance compared to the DAE.

3.3 Inference Hypotheses for Zero-Shot

In this section, we introduce two different inference hypotheses H_1, H_2 for the zero-shot of our approach. First, H_1 uses only a single-path from Enc_t to the decoder. Second, H_2 uses the weighted sum of the bi-path from both $Enc_{t'}$ and $Enc_{t''}$ to the decoder. Regarding hypothesis H_2, we use the same weights as those utilized during the DBAE training.

4 Experiments

4.1 Dataset

For the training and the experiments, we use *MultiUN*[1] (Ziemski et al., 2016). The *MultiUN* is in

[1]https://conferences.unite.un.org/UNCorpus

[1]The United Nations Parallel Corpus is made available without warranty of any kind, explicit or implied. The United Nations specifically makes no warranties or representations as to the accuracy or completeness of the information contained in the United Nations Corpus.

En->Any

ID	Model	1:1:1				1:2:4				1:5:50			
		Low	**Med.**	**High**	**Avg.**	**Low**	**Med.**	**High**	**Avg.**	**Low**	**Med.**	**High**	**Avg.**
①	Johnson et al. (2017)	41.6	43.2	39.5	41.8	41.4	43.5	40.2	42.0	39.4	44.5	41.6	41.9
②	Lyu et al. (2020)	46.3	46.2	43.4	45.7	46.4	47.1	44.6	46.3	41.1	47.8	45.9	44.8
③	Liao et al. (2021)	46.0	46.3	42.9	45.5	46.3	47.9	44.4	46.6	40.8	47.5	45.3	44.4
④	Purason and Tättar (2022)	46.1	46.8	43.1	45.8	46.3	47.4	44.2	46.3	41.1	47.7	45.5	44.6
⑤	**Ours**	**48.1**	**48.7**	**44.2**	**47.6**	**48.4**	**50.1**	**45.7**	**48.5**	**42.4**	**49.9**	**47.2**	**46.4**

Table 1: BLEU scores of En-to-Any *MultiUN* test set. "Low" denotes average BLEU when the target text is in either *Ar* or Zh; "Med." denotes average BLEU when the target text is in either *Es* or *Ru*; and "High" denotes average BLEU when target text is in *Fr*. "1:1:1" denotes *balanced* ratio. Both "1:2:4" and "1:5:50" present *imbalanced* ratio; every ratio represents "Low:Med.:High". For more detail of the model configuration, please refer to Appendix C. For more detail of the BLEU scores, please refer to Appendix D

.

Any->En

ID	Model	1:1:1				1:2:4				1:5:50			
		Low	**Med.**	**High**	**Avg.**	**Low**	**Med.**	**High**	**Avg.**	**Low**	**Med.**	**High**	**Avg.**
①	Johnson et al. (2017)	45.9	47.2	41.6	45.6	45.7	47.7	42.1	45.8	48.0	42.9	45.4	43.7
②	Lyu et al. (2020)	**52.4**	**53.3**	46.8	**51.6**	**52.6**	**54.2**	48.9	**52.5**	45.6	**55.1**	**50.6**	50.4
③	Liao et al. (2021)	51.3	53.0	47.4	51.2	51.3	54.1	48.9	51.9	46.1	54.6	49.9	50.3
④	Purason and Tättar (2022)	51.1	52.7	47.8	51.1	51.3	53.9	48.9	51.9	46.2	54.6	50.1	50.4
⑤	**Ours**	51.0	53.2	**48.3**	51.3	51.5	**54.2**	**49.3**	52.2	**46.9**	54.9	50.1	**50.7**

Table 2: BLEU scores of Any-to-En *MultiUN* test set. For more detail of the model configuration, please refer to Appendix C. For more detail of the BLEU scores, please refer to Appendix D

.

six languages (En, Ar, Zh, Fr, Ru, Es). And, for our work, we use the English-centric pairs out of the entire dataset. In line with Lyu et al. (2020), we manage the quantity of the dataset to establish two distinctive settings, balanced and unbalanced settings. Initially, we categorize all English-centric pairs into three distinct groups according to their quantities: *Low* (En <–> Ar, Zh), *Medium* (En <–> Es, Ru), and *High* (En <–> Fr).

To ensure that each language pair is not multi-parallel (Al-Shedivat and Parikh, 2019; Lyu et al., 2020), we gather mutually non-parallel English-centric pairs from multi-parallel corpus[1]. For experiments, we create three ratios by the amount of the English-centric data, which are as follows: (i) 1:1:1 (*Low*:*Medium*:*High*); (ii) 1:2:4 (*Low*:*Medium*:*High*); and (iii) 1:5:50 (*Low*:*Medium*:*High*).

Both the balanced (e.g., 1:1:1) and the unbalanced settings (e.g., 1:2:4, 1:5:50) are used for the training of the L_{mnmt}. For the DBAE task, we gather the monolingual corpus from the original multi-parallel data. We make use of both official *devset* and *testset* from MultiUN website[1]. As the *devset* and *testset* are multi-parallel, we use them for the English-centric result and the zero-shot. The *devset* and *testset* consist of 4K lines. More information regarding the dataset statistics used for the training can be found in Appendix B.

4.2 Training

For the fundamental model architecture, we use *Transformer*(Vaswani et al., 2017). Utilizing *TensorFlow Model Garden*[2] (Hongkun Yu and Li, 2020), we implement our model and baselines. All models for the experiments are executed for up to 100 epochs. Additionally, we set the batch size for the training to 16K tokens. Four NVIDIA Tesla V100 GPUs are used, distributing 4K tokens per GPU. Training our model takes roughly three days.

During the L_{mnmt}, we choose the best checkpoint by the minimum averaged *devset* loss. Unlike Gu et al. (2019), we only validate our model using the English-centric *devset*. When training the DBAE, we assign the weights of 0.5 to the output of $Enc_{t'}$ and another weight of 0.5 to $Enc_{t''}$.

Similar to Lyu et al. (2020), we preserve language-specific embeddings for the language-specific encoders/decoders. We establish the vocabulary size as 16K. With an embedding dimension of 512, we configure the feed-forward layer to have

[2]Licensed under the Apache License, Version 2.0

		Zero-Shot: Any->{Low,Med.,High}											
		1:1:1				1:2:4				1:5:50			
		Low	Med.	High	Avg.	Low	Med.	High	Avg.	Low	Med.	High	Avg.
①	**Liao et al. (2021) E.12 D.2**	33.7	35.3	34.7	34.5	33.9	35.8	34.8	34.8	32.2	35.0	33.5	33.6
②	**Ours DBAE(H_1)**	**36.9**	**36.3**	**35.8**	**36.4**	**36.9**	**36.3**	**35.9**	**36.5**	**34.0**	**35.7**	**35.5**	**35.0**
③	**Ours DBAE(H_2)**	34.8	35.7	35.3	35.3	34.8	35.9	35.5	35.4	31.6	35.2	35.0	33.7

Table 3: BLEU scores of zero-shot *MultiUN* testset. DBAE(H_1) stands for the zero-shot inference by H_1 hypothesis. DBAE(H_2) stands for the zero-shot inference by H_2 hypothesis. For more detail of the model configuration, please refer to Appendix C. For more detail of the BLEU scores, please refer to Appendix D.

		Zero-Shot: Any->{Low,Med.,High}											
		1:1:1				1:2:4				1:5:50			
		Low	Med.	High	Avg.	Low	Med.	High	Avg.	Low	Med.	High	Avg.
①	Johnson et al. (2017) Piv.	33.0	33.6	33.3	33.3	33.0	33.8	33.6	33.4	31.6	34.2	34.1	33.1
②	Liao et al. (2021)	33.4	34.8	33.5	34.0	33.5	34.3	34.3	34.0	32.0	34.1	33.5	33.1
③	**Ours DBAE(H_1)**	**36.9**	**36.3**	**35.8**	**36.4**	**36.9**	**36.3**	**35.9**	**36.5**	**34.0**	**35.7**	**35.5**	**35.0**
④	**Ours DAE(H_1)**	35.3	35.1	34.7	35.1	36.4	35.8	35.1	35.9	33.6	34.8	33.9	34.2

Table 4: BLEU scores of zero-shot *MultiUN* testset. DBAE(H_1) stands for the inference by H_1 hypothesis. "Piv." stands for the pivot translation. For more detail of the model configuration, please refer to Appendix C. For more detail of the BLEU scores, please refer to Appendix D.

		Zero-Shot: Any->{Low,Med.,High}											
		1:1:1				1:2:4				1:5:50			
		Low	Med.	High	Avg.	Low	Med.	High	Avg.	Low	Med.	High	Avg.
①	**DBAE(H_1) alph. 0.8**	34.5	34.5	34.2	34.4	35.4	34.8	35.0	35.1	31.2	33.8	33.5	32.7
②	**DBAE(H_1) alph. 0.6**	35.3	34.6	34.5	34.8	35.4	34.8	34.7	35.0	**32.9**	34.1	33.9	33.6
③	**DBAE(H_1) alph. 0.5**	**35.5**	35.0	**35.0**	**35.2**	35.5	34.9	**35.1**	35.2	32.6	**34.4**	**34.5**	**33.7**
④	**DBAE(H_1) alph. 0.4**	35.3	**35.1**	34.7	35.1	**35.7**	**35.2**	34.8	**35.3**	32.3	34.3	34.3	33.5
⑤	**DBAE(H_1) alph. 0.2**	35.0	34.3	33.7	34.5	35.6	**35.2**	34.9	**35.3**	32.6	34.1	**34.5**	33.6

Table 5: BLEU scores of zero-shot *MultiUN* devset. "alph." represents coefficient used for the Denoising Bi-path Autoencoder. For example, "alph. 0.8" stands for "α = 0.8" of $Enc_{t'}$. For more detail of the BLEU scores, please refer to Appendix D.

a size of 2048. For the learning rate scheduler, we follow the same learning scheduler as Vaswani et al. (2017): the learning rate linearly increases during *warm-up* step and then decreases proportionally to the square root of global step number after the *warm-up* step. We set the *warm-up* step to 8K. Moreover, we utilize the Adam optimizer with β_1=0.9, β_2=0.98, and ϵ=10^{-9}.

Sentencepiece[3] (Kudo and Richardson, 2018) is used to tokenize all sentences utilized for training and inference. To train the L_{mnmt}, we append the target language token (Johnson et al., 2017) to the beginning of the input sentences for all the methods in the experiments, except during the training of the DBAE or the DAE. For the unbalanced settings, we use *temperature sampling* with T = 5 to sample from the corpus (Arivazhagan et al., 2019).

To accelerate the training process, we set the layer count of the language-specific encoders to 10 and the language-specific decoders to 2 (Kong et al., 2022; Kasai et al., 2020; Bérard et al., 2021). For assigning the number of layers to the Enc_{all} (both for our model and the baselines), following the descriptions provided in Purason and Tättar (2022); Liao et al. (2021), we set two-thirds of the number of the language-specific encoder layers (truncating decimal points), e.g., 6 for this work. We describe the number of layers and parameters for the training and the inference across all experiments in Appendix C.

We perform an ablation study with varying numbers of layers for both the Enc_s and the Enc_t, and report the findings in Appendix A. To prevent *catastrophic forgetting* (Howard and Ruder, 2018) during the DBAE, we freeze Enc_s and Enc_{all} similar to how Ji et al. (2020) froze the initial layers of the encoder. It should be noted that the DBAE task is always run under a balanced setting when

[3]https://github.com/google/sentencepiece

[3]Licensed under the Apache License, Version 2.0 (the "License")

performed alternately with the L_{mnmt}.

4.3 Evaluation

In both the section 5.1 and 5.3, we evaluate all possible English-centric directions and non-English directions (the zero-shot). We also experiment with performance differences caused by the DAE or the DBAE. For the baselines on the zero-shot experiment, we try the zero-shot translation by Liao et al. (2021) and the pivot translation by Johnson et al. (2017).

For the section 5.1, neither all the baselines nor our proposed method go through the DAE and the DBAE during their training. For the section 5.3, the zero-shot baseline by Liao et al. (2021) goes through the DAE during the training. We also present our method trained by either the DAE or the DBAE. Since we prepend the pivot language token during training on Any–>En directions, we prepend the pivot language token to the input sentences for the zero-shot inference.

In section 5.2, we experiment with the zero-shot performance differences caused by the H_1 or the H_2 hypothesis. Since the H_2 hypothesis incorporates the bi-path, we add more encoder layers (equal to the number layers in $Enc_{t''}$) to the Liao et al. (2021) baseline to maintain the approximately same number of parameters.

In the section 5.4, we discuss how different weighted summations of $Enc_{t'}$ and $Enc_{t''}$ during the DBAE affect the zero-shot performance. Utilizing the devset, we validate whether our weight selection from the section 4.2 is the most approximately optimal for the zero-shot. As an evaluation metric, we use *SacreBLEU*[4] (Post, 2018); we use various tokenizer options such as *Zh* for Chinese and *13a* for the remaining languages.

5 Result

In this section, we evaluate our proposed method against other baselines. For all experiments, the model configurations are provided in Appendix C for all experiments. The complete BLEU scores for all the experiments in this section are presented in Appendix D.

5.1 English-centric Result

In Table 1, our method outperforms all the baseline methods for En–>Any directions. It is observed that our modularized encoder with Enc_t refines the interlingual features, resulting in improved performance.

In Table 2, it is shown that our approach does not exhibit significant improvements compared to the "En–>Any" directions. However, we notice that our encoder modules for the Any–>En directions have the same structure as those proposed by Liao et al. (2021), which yields comparable results. Thus, we deduce that our proposed method does not suffer from underfitting in "Any–>En" directions based on these observations.

The upper layers sharing of the language-specific encoders (③) exhibits poorer performance compared to our approach or another LS module approach (④). In this sense, this result supports the earlier argument, wherein the deficit language specific features lead to the performance degradation. Moreover, we infer that the explicit language-specific feature is useful.

The Full Layer Sharing (①) exhibits poorer performance compared to the other baselines including our proposed method. This is anticipated as the modularized MNMT approach utilizes three to five times more parameters during training.

Additionally, we expect that further developing the decoder might enhance Any–>En directions. However, the development of the decoder is out of the topic of our work, and we leave further development on this issue as future work.

5.2 Zero-Shot Hypotheses Result

Table 3 shows the zero-shot performance according to different zero-shot hypotheses. The hypothesis H_2 (③) outperforms Liao et al. (2021) (①) although they have almost the same capacity. From this, we can conclude that the zero-shot bi-path setting is still useful for the zero-shot.

Hypothesis H_1 (②) outperforms the hypothesis H_2. Even the H_1 hypothesis uses the single path setting having fewer parameters than the H_2 hypothesis. Through this experiment, we conclude that the H_1 hypothesis is good enough to be used for zero-shot inference.

5.3 Zero-Shot Result

From Table 4, we can see that our proposed method (③, ④) outperforms both the pivot (①) and the zero-shot translation (②). In particular, the DAE utilizing only $Enc_{t'}$ (④) demonstrates better performance than the one without any $Enc_{t'}$ (②). Based on this observation, we infer that the

[4]Licensed under the Apache License, Version 2.0 (the "License")

$Enc_{t'}$ itself aids in aligning the non-pivot language-specific encoders/decoders.

Moreover, across all data ratios, DBAE (③) shows enhanced zero-shot performance when compared to all other baselines. Of particular interest, in comparison to the DAE utilizing just $Enc_{t'}$ (④), we can infer that the output of $Enc_{t''}$ during the DBAE indeed helps align the non-pivot language-specific encoders/decoders.

As previously discussed in Section 4.2, the output of $Enc_{t'}$ is assigned equal weights as that of $Enc_{t''}$. Through this weight assignment, we can infer that the pivot language-specific feature is equally important as the target language-specific feature during the DBAE. To support our weight assignment, Section 5.4 discusses how varying weights on $Enc_{t'}$ and $Enc_{t''}$ during DBAE affect the zero-shot performance.

5.4 Weight Ablation of Encoder Modules during DBAE

Table 5 demonstrates that our proposed method achieves better zero-shot performance when both $Enc_{t''}$ and $Enc_{t'}$ are assigned equal weights in the context of DBAE. In contrast, if $Enc_{t'}$ is assigned a higher weight than $Enc_{t''}$, or vice versa, the resulting zero-shot performance deteriorates.

We can infer that during the alignment, the pivot language-specific feature is as important as the target language-specific feature. This supports the DBAE outperformance (compared to the DAE) observed in the section 5.3. And, it suggests the need of the DBAE for the zero-shot improvement.

6 Conclusion

In this work, we propose a new modularized MNMT with a modularized encoder consisting of the encoder modules of which each implements effective sharing strategies. For the zero-shot, we develop Denoising Bi-path Autoencoder (DBAE) to train the non-pivot language-specific decoders to be robust to the output by the non-pivot language.

In the experiment, we demonstrate that our modularized encoder enhances the overall performance of the English-centric tests. The DBAE also improves the zero-shot performance. Ultimately, we hope that our work sheds light on the modularized MNMT once more and believe that the modularized MNMT can serve as a path to the entrance of the optimal state of MNMT.

7 Limitations

In this work, we have tried 5 languages (paired with English in bi-direction) for our modularized MNMT. Similar to what Firat et al. (2016b) mentioned in their Limitation, a total of 10 English-centric pairs are not suffficient to thoroughly prove the optimal performance of our method.

We conduct experiments under simulated balanced and unbalanced settings, so our experiments do not take into account low-resource languages in the real world as reported by Goyal et al. (2022). Furthermore, for evaluation purposes, we refrain from using other publicly available evaluation benchmarks like *FLORES-101* (Goyal et al., 2022), *WMT*[5] . As a result, our experiments fail to account for diverse domains.

Despite being evaluated across a limited number of languages, our languages represent distinct language families (e.g., European, Arabic, Chinese). Given that our model performs well across these disparate language families, which is unfavorable for the creation of an interlingua, our findings suggest that modularized MNMT possesses greater potential and robustness

Our work demonstrates significant improvement in En–>Any but not in Any–>En directions. Future work will be required to enhance the performance of Any->En. Additionally, we do not implement incremental learning (Liao et al., 2021; Lyu et al., 2020) specifically designed for our approach.

For the DBAE, we do not develop an expert module (Shazeer et al., 2017; Zhang et al., 2021) to obtain automatic weights for the weighted summation. As a result, it is possible that the α setting for the DBAE experiment is not ideal. Additionally, we do not experiment with various cases beyond the bi-path (e.g., tri-path, quadri-path, and so on) to observe performance changes.

In the future, we intend to incorporate more languages into our experiment through the development of incremental learning. This will enable us to overcome the issue of insufficient language kinds. Rather than relying on simulated data settings, we will make use of additional public evaluation benchmarks to better reflect the status of languages in the real world.

[5]https://www.statmt.org/wmt22/

References

Maruan Al-Shedivat and Ankur P Parikh. 2019. Consistency by agreement in zero-shot neural machine translation. *arXiv preprint arXiv:1904.02338*.

Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Dmitry Lepikhin, Melvin Johnson, Maxim Krikun, Mia Xu Chen, Yuan Cao, George Foster, Colin Cherry, et al. 2019. Massively multilingual neural machine translation in the wild: Findings and challenges. *arXiv preprint arXiv:1907.05019*.

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

Ankur Bapna, Naveen Arivazhagan, and Orhan Firat. 2019. Simple, scalable adaptation for neural machine translation. *arXiv preprint arXiv:1909.08478*.

Alexandre Bérard, Dain Lee, Stéphane Clinchant, Kweonwoo Jung, and Vassilina Nikoulina. 2021. Efficient inference for multilingual neural machine translation. *arXiv preprint arXiv:2109.06679*.

Graeme Blackwood, Miguel Ballesteros, and Todd Ward. 2018. Multilingual neural machine translation with task-specific attention. *arXiv preprint arXiv:1806.03280*.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Raj Dabre, Chenhui Chu, and Anoop Kunchukuttan. 2020. A survey of multilingual neural machine translation. *ACM Computing Surveys (CSUR)*, 53(5):1–38.

Daxiang Dong, Hua Wu, Wei He, Dianhai Yu, and Haifeng Wang. 2015. Multi-task learning for multiple language translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1723–1732.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016a. Multi-way, multilingual neural machine translation with a shared attention mechanism. *arXiv preprint arXiv:1601.01073*.

Orhan Firat, Baskaran Sankaran, Yaser Al-Onaizan, Fatos T Yarman Vural, and Kyunghyun Cho. 2016b. Zero-resource translation with multilingual neural machine translation. *arXiv preprint arXiv:1606.04164*.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The Flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.

Jiatao Gu, Yong Wang, Kyunghyun Cho, and Victor OK Li. 2019. Improved zero-shot neural machine translation via ignoring spurious correlations. *arXiv preprint arXiv:1906.01181*.

Xianzhi Du Yeqing Li Abdullah Rashwan Le Hou Pengchong Jin Fan Yang Frederick Liu Jaeyoun Kim Hongkun Yu, Chen Chen and Jing Li. 2020. TensorFlow Model Garden. `https://github.com/tensorflow/models`.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*.

Baijun Ji, Zhirui Zhang, Xiangyu Duan, Min Zhang, Boxing Chen, and Weihua Luo. 2020. Cross-lingual pre-training based transfer for zero-shot neural machine translation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 115–122.

Renren Jin and Deyi Xiong. 2022. Informative language representation learning for massively multilingual neural machine translation. *arXiv preprint arXiv:2209.01530*.

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

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1700–1709.

Jungo Kasai, Nikolaos Pappas, Hao Peng, James Cross, and Noah A Smith. 2020. Deep encoder, shallow decoder: Reevaluating non-autoregressive machine translation. *arXiv preprint arXiv:2006.10369*.

Xiang Kong, Adithya Renduchintala, James Cross, Yuqing Tang, Jiatao Gu, and Xian Li. 2022. Multilingual neural machine translation with deep encoder and multiple shallow decoders. *arXiv preprint arXiv:2206.02079*.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*.

Junwei Liao, Yu Shi, Ming Gong, Linjun Shou, Hong Qu, and Michael Zeng. 2021. Improving zero-shot neural machine translation on language-specific encoders-decoders. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Yichao Lu, Phillip Keung, Faisal Ladhak, Vikas Bhardwaj, Shaonan Zhang, and Jason Sun. 2018. A neural interlingua for multilingual machine translation. *arXiv preprint arXiv:1804.08198*.

Minh-Thang Luong, Quoc V Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2015a. Multi-task sequence to sequence learning. *arXiv preprint arXiv:1511.06114*.

Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015b. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.

Sungwon Lyu, Bokyung Son, Kichang Yang, and Jaekyoung Bae. 2020. Revisiting Modularized Multilingual NMT to Meet Industrial Demands. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5905–5918, Online. Association for Computational Linguistics.

Jerin Philip, Alexandre Berard, Matthias Gallé, and Laurent Besacier. 2020. Monolingual adapters for zero-shot neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4465–4470.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Taido Purason and Andre Tättar. 2022. Multilingual neural machine translation with the right amount of sharing. In *Proceedings of the 23rd Annual Conference of the European Association for Machine Translation*, pages 91–100, Ghent, Belgium. European Association for Machine Translation.

Zhi Qu and Taro Watanabe. 2022. Adapting to non-centered languages for zero-shot multilingual translation. *arXiv preprint arXiv:2209.04138*.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27.

Ahmet Üstün, Alexandre Berard, Laurent Besacier, and Matthias Gallé. 2021. Multilingual unsupervised neural machine translation with denoising adapters. *arXiv preprint arXiv:2110.10472*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Raúl Vázquez, Alessandro Raganato, Jörg Tiedemann, and Mathias Creutz. 2018. Multilingual nmt with a language-independent attention bridge. *arXiv preprint arXiv:1811.00498*.

Fei Yuan, Yinquan Lu, Wenhao Zhu, Lingpeng Kong, Lei Li, Yu Qiao, and Jingjing Xu. 2023. Lego-MT: Learning detachable models for massively multilingual machine translation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 11518–11533, Toronto, Canada. Association for Computational Linguistics.

Biao Zhang, Ankur Bapna, Rico Sennrich, and Orhan Firat. 2021. Share or not? learning to schedule language-specific capacity for multilingual translation.

Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. 2020. Improving massively multilingual neural machine translation and zero-shot translation. *arXiv preprint arXiv:2004.11867*.

Changfeng Zhu, Heng Yu, Shanbo Cheng, and Weihua Luo. 2020. Language-aware interlingua for multilingual neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1650–1655.

Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The united nations parallel corpus v1. 0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3530–3534.

A Ablation for Number of Layers

Assuming we fix the number of Enc_{all} layers to the certain ratio of the number of encoder layers (e.g., two-thirds of the encoder layers), we have variations of the number of Enc_s, Enc_t layers. Table 10,11, and 12 present the performance variation according to the number of Enc_s, Enc_t layers. Each ratio column represents the average BLEU according to the data amount ratio.

We conduct the ablation study under the hypothesis H_1 for the zero-shot. In Table 10 and 11, we observe the improvement when Enc_t has more number layers than the Enc_s. However, in Table 12, the zero-shot improvement does not hold when the Enc_t has more number layers than the Enc_s. In order to prevent further variance, we set the number of both Enc_t and Enc_s layers equal for the main experiments.

		1:1:1	1:2:4	1:5:50
High	**En<–>Fr**	2,200,000	4,400,000	8,900,000
Medium	**En<–>Ru**	2,200,000	2,200,000	875,000
	En<–>Es	2,200,000	2,200,000	875,000
Low	**En<–>Zh**	2,200,000	1,100,000	175,000
	En<–>Ar	2,200,000	1,100,000	175,000

Table 6: Dataset statistic for *MultiUN* dataset.

B Dataset Statistic

Table 6 shows the number of lines of each English-centric corpus. Each ratio column represents the number of lines distributed by different ratios (e.g., 1:1:1, 1:2:4, and 1:5:50). The ratio represents "Low": "Med.": "High". The total number of lines on each column is 11M lines. Each multi-parallel monolingual corpus, consisting of 11M lines, is used for both the DAE and the DBAE tasks.

C Model Configuration

Table 7, 8, and 9 describe the number of layers of the encoder (e.g., Enc #), the encoder modules (e.g., Enc_s #, Enc_{all} #, $Enc_{t'}$ #, $Enc_{t''}$ #, Enc_t #) and the decoder (e.g., Dec #). They also show the number of parameters of all experimenting models during the training and the inference. Table 7 corresponds to the configuration of Table 1, 2. Table 8 corresponds to the configuration of Table 4. Table 9 corresponds to the configuration of Table 3.

D Full BLEU Scores

As we do not show the full BLEU scores of the experiments in Section 5, we instead provide them in this Appendix. For Table 1, refer to Table 13. Refer to Table 14 for the full BLEU scores of Table 2. Check Table 15 for the full BLEU scores of Table 3. For Table 4, Table 16 displays the full BLEU scores. Finally, for Table 5, Table 17 displays the full BLEU scores.

ID	Model	Enc #	Enc_s #	Enc_{all} #	Enc_g #	Enc_t #	Dec #	Train Param. #	Infer. Param. #
①	Johnson et al. (2017)	10	–	–	–	–	2	56,289,280	56,289,280
②	Lyu et al. (2020)	10	–	–	–	–	2	288,583,680	56,289,280
③	Liao et al. (2021)	10	4	6	–	–	2	194,079,744	56,289,280
④	Purason and Tättar (2022)	10	2	6	2	–	2	187,778,048	56,289,280
⑤	Ours	10	2	6	–	2	2	194,079,744	56,289,280

Table 7: Model configuration for En-centric evaluation. Each column represents as follows: Enc #: The number of encoder layers; Enc_s#: The number of layers in Enc_s module; Enc_{all} #: The number of layers in Enc_{all} module; Enc_g #: The number of layers in Enc_g module; Enc_t #: The number of layers in Enc_t module; Dec #: The number of decoder layers; ; Train Param. #: The total number of parameters of all encoder/decoder modules involved during the training; Infer. Param. #: The number of parameters used during inference.

ID	Model	Enc #	Enc_s #	Enc_{all} #	Enc_g #	Enc_t #	Dec #	Train Param. #	Infer. Param. #
①	Johnson et al. (2017) Piv.	10	–	–	–	–	2	56,289,280	56,289,280
②	Liao et al. (2021) E.10 D.2	10	4	6	–	–	2	194,079,744	56,289,280
③	Ours w/ DBAE(H_1)	10	2	6	–	2	2	194,079,744	56,289,280
④	Ours w/ DAE(H_1)	10	2	6	–	2	2	194,079,744	56,289,280

Table 8: Model configuration for the zero-shot evaluation.

ID	Model	Enc #	Enc_s #	Enc_{all} #	$Enc_{t'}$ #	$Enc_{t''}$ #	Enc_t	Dec #	Train Param. #	Infer. Param. #
①	Liao et al. (2021) E.12 D.2	12	4	8	–	–	–	2	200,380,416	62,589,952
②	Ours w/ DBAE(H_1)	10	2	6	–	–	2	2	194,079,744	56,289,280
③	Ours w/ DBAE(H_2)	10	2	6	2	2	–	2	194,079,744	62,590,976

Table 9: Model configuration for the zero-shot inference hypotheses (H_1,H_2) evaluation.

En–>Any							
Enc #	Enc_s #	Enc_{all} #	Enc_t #	**Dec #**	**1:1:1**	**1:2:4**	**1:5:50**
10	2	6	2	2	47.7	48.1	47.5
10	3	6	1	2	46.8	47.5	46.8
10	1	6	3	2	**47.9**	**48.4**	**47.9**

Table 10: Ablation study of the number of layers in the encoder module under En–>Any evaluation.

Any–>En							
Enc #	Enc_s #	Enc_{all} #	Enc_t #	**Dec #**	**1:1:1**	**1:2:4**	**1:5:50**
10	2	6	2	2	50.3	51.1	50.5
10	3	6	1	2	50.3	50.9	**50.7**
10	1	6	3	2	**51.1**	**51.9**	50.4

Table 11: Ablation study of the number of layers in the encoder module under Any–>En evaluation.

Zero-shot Inference							
Enc #	Enc_s **#**	Enc_{all} **#**	Enc_t **#**	**Dec #**	**1:1:1**	**1:2:4**	**1:5:50**
10	2	6	2	2	35.8	**36.3**	34.0
10	3	6	1	2	**36.0**	35.9	34.0
10	1	6	3	2	35.7	36.1	**34.6**

Table 12: Ablation study of the number of layers in the encoder module under the zero-shot evaluation.

	1:1:1					1:2:4					1:5:50				
	①	②	③	④	⑤	①	②	③	④	⑤	①	②	③	④	⑤
En-Fr	39.5	43.4	42.9	43.1	44.2	40.2	44.6	44.4	44.2	45.7	41.6	45.9	45.3	45.5	47.2
En-Ru	35.0	38.4	38.6	39.1	40.9	35.3	39.2	39.9	39.4	42.1	35.6	39.8	39.3	39.5	41.6
En-Es	51.3	53.9	54.0	54.5	56.6	51.8	55.0	55.9	55.4	58.1	53.4	55.8	55.6	55.8	58.3
En-Zh	47.4	53.7	53.2	53.0	53.8	47.5	53.9	53.5	53.6	54.2	46.3	50.2	49.3	49.8	49.8
En-Ar	35.8	38.9	38.9	39.3	42.4	35.3	38.8	39.2	39.1	42.5	32.6	32.1	32.4	32.3	35.1
Average	41.8	45.7	45.5	45.8	47.6	42.0	46.3	46.6	46.3	48.5	41.9	44.8	44.4	44.6	46.4

Table 13: A full list of the BLEU score of Table 1. ①: Johnson et al. (2017); ②: Lyu et al. (2020); ③: Liao et al. (2021); ④: Purason and Tättar (2022); ⑤: Ours

	1:1:1					1:2:4					1:5:50				
	①	②	③	④	⑤	①	②	③	④	⑤	①	②	③	④	⑤
Fr-En	41.6	46.8	47.4	47.8	48.3	42.1	48.9	48.9	48.9	49.3	42.9	50.6	49.9	50.1	50.1
Ru-En	43.7	48.6	48.7	48.3	48.8	44.0	49.7	49.6	49.5	49.8	44.4	50.4	50.0	50.2	50.4
Es-En	50.8	58.0	57.3	57.1	57.6	51.3	58.7	58.7	58.4	58.7	51.5	59.7	59.2	59.1	59.5
Zh-En	43.4	49.0	48.0	47.9	47.8	43.6	49.7	47.9	48.1	48.5	42.4	44.0	43.3	43.8	44.2
Ar-En	48.4	55.8	54.6	54.3	54.2	47.8	55.6	54.7	54.5	54.6	45.9	47.3	48.9	48.6	49.6
Average	43.7	51.6	51.2	51.1	51.3	45.8	52.5	51.9	51.9	52.2	45.4	50.4	50.3	50.4	50.7

Table 14: A full list of the BLEU score of Table 2. ①: Johnson et al. (2017); ②: Lyu et al. (2020); ③: Liao et al. (2021); ④: Purason and Tättar (2022); ⑤: Ours

	1:1:1			1:2:4			1:5:50		
	①	②	③	①	②	③	①	②	③
Fr-Es	42.7	43.9	43.2	43.4	44.5	43.8	44.2	44.7	44.0
Fr-Zh	38.8	42.5	40.02	38.8	42.8	40.3	39.1	40.9	37.3
Fr-Ar	26.4	29.1	27.6	27.2	28.9	27.7	25.6	26.8	25.2
Fr-Ru	30.4	31.4	30.63	31.0	31.3	30.6	30.9	32.1	30.9
Es-Fr	40.7	43.0	42.1	40.6	43.0	42.4	40.1	43.6	42.4
Es-Zh	41.9	47.1	44.1	41.7	47.1	44.4	42.0	44.8	40.7
Es-Ar	30.4	34.3	32.6	30.9	34.1	32.6	28.2	31.1	29.4
Es-Ru	33.5	35.7	34.6	34.1	35.8	34.6	33.9	36.3	34.9
Zh-Fr	28.2	28.3	28.4	27.9	28.2	28.3	25.6	27.1	27.0
Zh-Es	33.3	35.3	35.3	34.2	34.7	35.1	32.7	33.2	33.2
Zh-Ar	24.3	25.6	24.6	24.2	25.4	24.4	20.7	21.4	21.2
Zh-Ru	26.5	26.1	26.2	26.7	25.9	25.9	24.7	24.5	24.8
Ar-Fr	35.5	37.1	36.5	35.8	37.2	37.0	33.8	35.0	34.6
Ar-Es	43.0	44.2	43.8	43.4	44.6	44.2	41.3	41.6	41.4
Ar-Zh	40.5	44.4	41.6	40.2	44.2	42.1	37.6	39.0	35.5
Ar-Ru	32.1	32.7	32.3	32.1	32.3	32.0	30.6	30.6	30.1
Ru-Fr	34.5	35.0	34.3	35.0	35.3	34.4	34.4	36.6	35.9
Ru-Es	40.6	40.8	39.9	41.6	41.2	40.6	41.8	42.8	42.2
Ru-Zh	39.9	43.0	40.2	40.0	43.7	40.4	39.3	41.4	38.1
Ru-Ar	27.6	28.8	27.4	27.9	28.7	26.9	25.0	26.6	25.2
Average	34.5	36.4	35.3	34.8	36.5	35.4	33.6	35.0	33.7

Table 15: A full list of the BLEU score of Table 3. ①: Liao et al. (2021) E.12 D.2; ②: Ours DBAE (H_1); ③: Ours DBAE (H_2)

	1:1:1				1:2:4				1:5:50			
	①	②	③	④	①	②	③	④	①	②	③	④
Fr-Es	39.0	42.7	43.9	42.4	39.9	42.2	44.5	43.2	40.8	43.6	44.7	45.0
Fr-Zh	37.0	38.4	42.5	41.3	37.8	39.2	42.8	42.6	37.1	39.2	40.9	41.6
Fr-Ar	25.2	26.7	29.1	27.9	25.7	26.4	28.9	28.5	24.4	25.4	26.8	26.9
Fr-Ru	27.1	29.8	31.4	31.2	27.7	29.5	31.3	31.2	27.8	30.0	32.1	31.2
Es-Fr	36.6	39.5	43.0	40.8	37.8	40.1	43.0	41.4	38.3	40.4	43.6	41.9
Es-Zh	41.3	41.9	47.1	44.9	41.9	42.4	47.1	46.6	40.9	41.4	44.8	44.9
Es-Ar	29.4	30.4	34.3	31.8	29.7	29.9	34.1	32.8	27.5	28.5	31.1	30.3
Es-Ru	30.5	32.9	35.7	35.1	31.2	32.6	35.8	35.1	31.0	32.8	36.3	34.9
Zh-Fr	31.3	26.6	28.3	27.9	32.1	27.9	28.2	28.6	31.9	25.9	27.1	25.5
Zh-Es	38.1	33.1	35.3	33.2	38.9	33.1	34.7	35.0	38.8	31.7	33.2	31.8
Zh-Ar	26.8	23.6	25.6	24.5	27.1	23.3	25.4	25.1	24.7	20.5	21.4	20.7
Zh-Ru	27.8	25.8	26.1	25.3	28.3	25.7	25.9	25.8	27.4	24.1	24.5	23.4
Ar-Fr	33.9	34.3	37.1	34.8	34.7	34.7	37.2	36.1	34.4	33.2	35.0	34.4
Ar-Es	41.4	42.3	44.2	41.6	42.4	41.5	44.6	43.6	41.5	40.3	41.6	41.8
Ar-Zh	41.5	39.6	44.4	41.3	41.9	40.5	44.2	44.1	39.7	36.7	39.0	39.7
Ar-Ru	30.2	31.1	32.7	31.3	30.4	30.0	32.3	32.2	29.6	29.1	30.6	30.1
Ru-Fr	32.6	33.7	35.0	35.3	33.5	34.6	35.3	34.4	34.2	34.6	36.6	33.9
Ru-Es	39.2	40.4	40.8	41.0	40.2	40.3	41.2	40.6	40.7	41.0	42.8	40.5
Ru-Zh	39.4	39.5	43.0	42.1	40.0	40.1	43.7	43.2	38.9	39.4	41.4	40.2
Ru-Ar	26.8	27.2	28.8	28.4	27.2	26.6	28.7	28.1	25.3	24.6	26.6	24.7
Average	33.3	34.0	36.4	35.1	33.4	34.0	36.5	35.9	33.1	33.1	35.0	34.2

Table 16: A full list of the BLEU score of Table 4. ①: Johnson et al. (2017) Piv.; ②: Liao et al. (2021); ③: Ours DBAE(H_1); ④: Ours DAE(H_1)

	1:1:1					1:2:4					1:5:50				
	①	②	③	④	⑤	①	②	③	④	⑤	①	②	③	④	⑤
Fr-Es	45.1	43.9	43.9	44.6	43.9	44.6	45.2	44.5	45.4	44.6	46.1	44.3	44.7	45.8	44.9
Fr-Zh	41.5	42.4	42.5	42.8	42.2	42.9	43.5	42.8	43.4	43.4	40.0	41.4	40.9	41.4	41.9
Fr-Ar	29.3	29.2	29.1	29.2	29.3	29.3	29.7	28.9	29.7	29.4	26.6	27.3	26.8	26.8	27.3
Fr-Ru	32.2	31.6	31.4	32.1	30.9	31.7	32.1	31.3	32.1	32.4	31.9	31.8	32.1	32.8	32.7
Es-Fr	42.9	42.8	43.0	41.7	41.0	42.7	42.6	43.0	42.6	41.7	43.3	42	43.6	43.3	42.8
Es-Zh	45.7	47.1	47.1	45.8	46.3	46.8	47.2	47.1	47.1	47	43.0	45.1	44.8	44.8	44.9
Es-Ar	33.8	34.3	34.3	33.5	33.3	33.7	33.3	34.1	33.4	33.2	30.6	31.2	31.1	30.2	30.2
Es-Ru	36.1	35.8	35.7	35.6	34.4	35.3	35.5	35.8	35.5	35.9	35.4	35.4	36.3	36.0	35.8
Zh-Fr	25.3	27.7	28.3	28.3	26.7	27.9	28.0	28.2	28.6	27.9	23.8	26.2	27.1	26.2	27.5
Zh-Es	32.6	33.6	35.3	34.5	33.8	33.1	33.8	34.7	34.8	34.5	30.9	31.4	33.2	31.8	31.8
Zh-Ar	22.9	25.3	25.6	25.3	24.8	25.2	24.8	25.4	25.3	24.9	19.2	20.8	21.4	20.3	20.5
Zh-Ru	24.3	25.9	26.1	25.8	25.3	25.4	25.3	25.9	26.0	26.0	22.1	24.3	24.5	23.1	24.2
Ar-Fr	35.5	35.8	37.1	36.6	36.2	37.3	36.1	37.2	35.5	36.7	35.2	35.5	35.0	34.6	35.2
Ar-Es	42.8	43.1	44.2	43.8	43.4	43.9	43.2	44.6	43.0	43.8	42.7	42.5	41.6	41.2	40.9
Ar-Zh	42.1	44.1	44.4	44.1	44.0	44.0	43.8	44.2	43.9	44.6	37.7	39.7	39.0	39.6	40.4
Ar-Ru	31.8	31.9	32.7	33.3	32.5	31.8	31.8	32.3	32.3	33.3	30.6	31.2	30.6	30.3	30.9
Ru-Fr	36.5	35.1	35.0	35.90	34.2	35.5	35.7	35.3	36.2	36.3	35.7	36.3	36.6	36.7	37.2
Ru-Es	42.3	41.0	40.8	41.7	40.2	42.2	41.4	41.2	42.9	41.9	41.3	42.7	42.8	42.8	42.1
Ru-Zh	43.0	43.3	43.0	43.0	43.0	43.2	43.3	43.7	44.5	44.4	39.1	42.3	41.4	42.0	42.0
Ru-Ar	29.7	29.0	28.8	30.2	28.8	29.0	28.7	28.7	29.7	29.8	25.1	27.1	26.6	25.6	26.6
Average	34.4	34.8	35.2	35.1	34.5	35.1	35.0	35.2	35.3	35.3	32.7	33.6	33.7	33.5	33.6

Table 17: A full list of the BLEU score of Table 5. ①: DBAE(H_1) alph. 0.8.; ②: DBAE(H_1) alph. 0.6; ③: DBAE(H_1) alph. 0.5; ④: DBAE(H_1) alph. 0.4; ⑤: DBAE(H_1) alph. 0.2