Adapting to Non-Centered Languages for Zero-shot Multilingual Translation

Zhi Qu and **Taro Watanabe**
Nara Institute of Science and Technology
{qu.zhi.pv5, taro}@is.naist.jp

Abstract

Multilingual neural machine translation can translate unseen language pairs during training, i.e. zero-shot translation. However, the zero-shot translation is always unstable. Although prior works attributed the instability to the domination of central language, e.g. English, we supplement this viewpoint with the strict dependence of non-centered languages. In this work, we propose a simple, lightweight yet effective language-specific modeling method by adapting to non-centered languages and combining the shared information and the language-specific information to counteract the instability of zero-shot translation. Experiments with Transformer on IWSLT17, Europarl, TED talks, and OPUS-100 datasets show that our method not only performs better than strong baselines in centered data conditions but also can easily fit non-centered data conditions. By further investigating the layer attribution, we show that our proposed method can disentangle the coupled representation in the correct direction.[1]

1 Introduction

Training multilingual neural machine translation (MNMT) system requires enormous number of parameters and resources, but the zero-shot translation, namely translating unseen language pairs during training, has shown the potential to simplify the MNMT (Firat et al., 2017). Johnson et al. (2017) has shown that adding language tokens, e.g. <en>, at the beginning of a sentence allows the model to build cross-linguistic representation by treating the token as translation instruction specifying target language. However, the zero-shot translation is always unstable. One possibility causing the instability of zero-shot translation is spurious correlation (Gu et al., 2019). The target linguistic representation captured by the model is directly and strictly dependent on encoded source linguistic information instead of learning specific representations for source and target language, then combining independent linguistic representations to generate results. Prior works (Lakew et al., 2019; Fan et al., 2020; Rios et al., 2020; Freitag and Firat, 2020; Liu et al., 2021) indicated that the spurious correlation is caused by the centered data condition in which multilingual data is constructed by bridging a central language, e.g. English, to other non-centered languages. The central language will dominate the representation in the MNMT model to degenerate the information specific to non-centered languages since multilingual data comprises a set of bilingual data constructed by coupling non-centered languages with the central language. However, the non-centered data condition without any central language is also unstable in zero-shot translation.[2] Therefore, simply attributing the instability of zero-shot translation to the central language cannot fit all cases of zero-shot translation.

We move the perspective from the domination of the central language to the weakness of non-centered languages. The problems of zero-shot translation could be attributed to the strict dependence of non-centered languages. Specifically, a non-centered language would strictly depend on another language as a strongly related language pair to prohibit learning robust and independent translation instructions for zero-shot translation. Under this hypothesis, the centered data condition is a special case of this description, because all non-centered languages depend on the central language. In this light, a key to improving zero-shot translation is disentangling non-centered languages from the strict dependence which is built in training.

Specifically, we model extra language-specific (LS) components (Sachan and Neubig, 2018; Philip et al., 2020; Escolano et al., 2021; Zhang et al., 2021) adapting to non-centered languages in a

[1] Codes and detailed results are available in: https://github.com/zhiqu22/AdapNonCenter

[2] We give specific examples in Section 4.1.

Proceedings of the 29th International Conference on Computational Linguistics, pages 5251–5265
October 12–17, 2022.

mixing shared and LS information mode (Zhang et al., 2021), our objective is to enhance the weak representations for assisting the balance of cross-linguistic representation in shared information container to improve the quality of translation Cheng et al. (2022); Shao and Feng (2022). Furthermore, the mixing mode can decrease the complexity of LS modeling since we treat the representation space of the MNMT model as the combination of shared and LS information, and we no longer build the independent representation space for each language (Sachan and Neubig, 2018; Escolano et al., 2021). In this motivation, we propose a simple, lightweight yet effective method to augment feed-forward network of Transformer (Vaswani et al., 2017) by LS components adapting to non-centered languages.

Our contributions are as follows:

- Our lightweight method achieves considerable gains on multilingual and zero-shot translation and performs stably in IWSLT17, Europarl, TED talks and OPUS-100.
- We describe the strict dependence of non-centered languages to supplement the prior viewpoint of zero-shot translation, and verify it by experiments under different data conditions with and without the central language.
- Our work explores decreasing complexity in LS modeling. We also through the analysis via layer attribution (Dhamdhere et al., 2019) to show the significance of our methods in decoupling representations of MNMT.

2 Related Work

Initially, Johnson et al. (2017) laid the foundation of zero-shot translation which endorses training the MNMT model under the centered data condition and put forward the thinking about the instability of data conditions on zero-shot translation. On this basis, Gu et al. (2019) also showed that the performance of zero-shot translation is sensitive to parameters for initialization, which is another cause of instability. In this paper, we systematically described this instability (Section 5.1) and tested it experimentally.

In the early stages, Mattoni et al. (2017) pointed out that increasing corpus size can effectively improve zero-shot translation. However, the spurious correlation (Gu et al., 2019) means that unreasonably increasing training data could degenerate the zero-shot translation due to strict dependence. Since then, the concern of the centered data condition was started to be discussed by several different strategies. Fan et al. (2020); Freitag and Firat (2020) augmented the training data to make all languages interconnected, which will result in an excessive increase in training costs. Lakew et al. (2019) explored incrementally training the MNMT model by monolingual data, and Gu et al. (2019); Zhang et al. (2020) generated synthetic data for the zero-shot directions by backtranslation. These methods transformed the zero-shot task to zero-resource task.

Another line of work on improving zero-shot translation is to adjust the learning of representations in the MNMT model. Lu et al. (2018); Pham et al. (2019); Zhang et al. (2020); Liu et al. (2021) focused on restricting the representation of encoder outputs to be language-agnostic, but the restriction may reduce the performance of the model trained by large-scale datasets. Pan et al. (2021) aligned representations from different languages via contrastive learning and the additional dictionary. Philip et al. (2020); Yang et al. (2021); Zhang et al. (2021) explored to enhance the influence of LS features in the translation. Our work continues in this direction, but with a special focus on only enhancing the decoding step and mixing shared and LS information.

Our work is based on LS modeling which is the heuristic variation of Mixture-of-Experts model (Shazeer et al., 2017), because it aims to build extra components as experts to directionally improve linguistic features. Sachan and Neubig (2018) and Escolano et al. (2021) built LS encoder or decoder, but multi-encoder/decoder architecture has too many parameters. Wang et al. (2018) divided neural cells into LS parts and Lin et al. (2021) divided LS subnets from the model, but these methods limited the learning capacity. Bapna and Firat (2019) and Philip et al. (2020) added LS adapters on the end of encoder and decoder and fine tuned for LS representations. Zaremoodi et al. (2018); Zhang et al. (2021) explored the paradigm of constructing LS components to assist the shared information. However, extra components always increase the cost of modeling significantly when languages existed too much. The investigation about the importance of LS information specified to target language (Lee et al., 2017; Blackwood et al., 2018; Pham et al., 2019; Wu et al., 2021) enlightens us to limit the

improving LS information in the decoding process to achieve lightweight LS modeling.

3 Central Language Aware Multilingual Neural Machine Translation

We employ Transformer (Vaswani et al., 2017) as the backbone to construct our architecture. Consider a set of m languages $\mathbb{L} = \{l_1, l_2, \ldots, l_m\}$, we assign the first language l_1 as the central language l_c. The non-centered set is the subset of $\mathbb{L}$, that is $\mathbb{L}' = \{l_2, l_3, \ldots, l_m\}$. We follow prior works (Zhang et al., 2020; Liu et al., 2021; Wu et al., 2021) to assign English as the center of multilingual data. Given the original input sequence of symbol representation to the encoder $\boldsymbol{x} = x_1, x_2, ..., x_i$ and the output sequence generated by decoder $\boldsymbol{y} = y_1, y_2, ..., y_j$, we follow the method of Johnson et al. (2017) to insert the language token at the beginning of $\boldsymbol{x}$ as translation instruction. Therefore, the actual input sequence is $\boldsymbol{x}' = (l, \boldsymbol{x})$, and we model the translation of $\boldsymbol{x}'$ to $\boldsymbol{y}$ with Transformer. We only build LS layers (LSLs) parallel with the Feed-Forward Network (FFN) layers in the decoder of Transformer, and keep the self-attention and cross-attention mechanism fixed.

The FFN of transformer consists of two fully connected neural networks with a ReLU activation function in between:

$$\mathrm{FFN}(\boldsymbol{h}) = \max(0, \boldsymbol{h}\mathbf{W}_1 + \boldsymbol{b}_1)\mathbf{W}_2 + \boldsymbol{b}_2 \quad (1)$$

Where $\boldsymbol{h}$ is the input vector, $\mathbf{W}$ indicates parameter matrices for projections, and $\boldsymbol{b}$ indicates bias parameter matrices. LSLs are a series of neural networks specified to $\mathbb{L}'$. Each LSL is similar to FFN in architecture but can be relatively light in inner size:

$$\mathrm{LSL}_l(\boldsymbol{h}) = \max(0, \boldsymbol{h}\mathbf{W}_1^l + \boldsymbol{b}_1^l)\mathbf{W}_2^l + \boldsymbol{b}_2^l \quad (2)$$

where $l \in \mathbb{L}'$. The trade-off between shared and LS information is difficult (Zhang et al., 2021; Wang and Zhang, 2021), because the information that each language carries is not absolutely equal. To balance the shared and LS information, we introduce a set of learnable scalars in each decoder layer $\mathbb{T} = \{t_{l_2}, t_{l_3}, \ldots, t_{l_m}\}$. Elements of $\mathbb{T}$ correspond to languages of $\mathbb{L}'$ one by one, then each t connects LS information with the shared information. We initialize t to 0.1, then parameters[3] of $\mathbb{T}$ are updated during training together with other parameters.

[3] we report the distribution of LS information weights for large-scale dataset (99 languages) in the Appendix C.

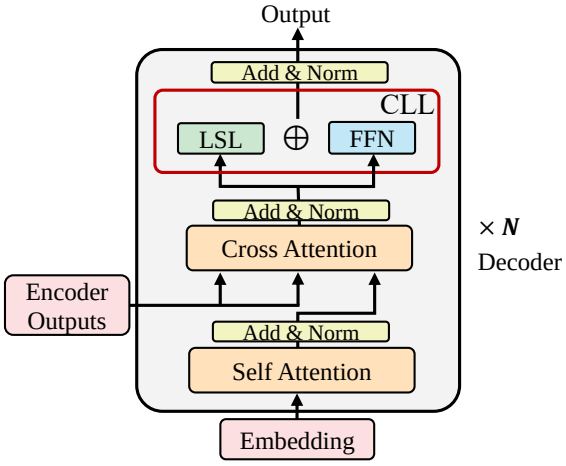


Figure 1: Illustrations of our proposed architecture modified from the decoder of Transformer. $\bigoplus$ indicates weighted plus, N is the number of decoder layers.

To differentiate the central language from non-centered languages, the original FFN of Transformer's decoder is used as the shared information space for all languages of $\mathbb{L}$, and we only construct lightweight LSLs to learn independent linguistic information for non-centered languages of $\mathbb{L}'$. Therefore, the complete architecture of **C**entral **L**anguage-aware **L**ayer (CLL) is:

$$\mathrm{CLL}_l(\boldsymbol{h}) = \begin{cases} \mathrm{FFN}(\boldsymbol{h}) + t_l \, \mathrm{LSL}_l(\boldsymbol{h}) & l \in \mathbb{L}' \\ \mathrm{FFN}(\boldsymbol{h}) & l = l_c \end{cases} \quad (3)$$

Based on the piecewise function Eq.(3), the role of central language will be abandoned in non-centered data conditions, namely the case of $l = l_c$ will not be triggered. We illustrate the architecture of CLL in Figure 1: The CLL is a component, including lightweight LSLs and FFN, to replace the original role of FFN in each decoder layer of Transformer. Compared to the Mixture of Experts which is the generalization of the gating mechanism (Shazeer et al., 2017), a deterministic route specific to language replaces the gate in CLL. For convenience, we use **FCLL** (full CLL) to indicate that the model in which all decoder layers are constructed in the form of our proposed architecture.

We introduce a variation named **SD** that constructs CLL in a single decoder layer among all layers of Transformer, namely Single-Disentangled CLL. Inspired by the work of Liu et al. (2021), we remove the residual connection of FFN in a middle encoder layer to weaken the linguistic features of

Method	+Params	Position
baseline	None	None
FCLL	$\mathcal{O}(k)$	Decoder
SD	$\mathcal{O}(k/N)$	Decoder
Philip et al. (2020)	$\mathcal{O}(2k)$	All
Zhang et al. (2021)	$\mathcal{O}(5k)$	All
Sachan and Neubig (2018)	$\mathcal{O}(K)$	Decoder

Table 1: Number of parameters required for different LS modeling methods. N, k and K denote the number of encoder/decoder layers, parameters per LS layer, and parameters per encoder/decoder layer ($k \ll K$), respectively. Position indicates the position of a model to construct LS components.

encoding. To keep the balance between weakening encoding and improving decoding, we empirically build CLL in the middle decoder. Specifically, given N encoder and decoder layers of Transformer, we remove the residual connection of the FFN in the encoder and replace the FFN with CLL in the decoder at $N/2 + 1^{th}$ layer of both networks. Our experiments (Section 4.3) empirically show that SD has comparable performance with FCLL in small-scale datasets, although SD is more parameter-efficient than FCLL (Table 1).

4 Experiments

4.1 Dataset

We take IWSLT17 (Cettolo et al., 2017) and restrict 4 languages from MMCR4NLP (Dabre and Kurohashi, 2019) to verify basic abilities of multilingual and zero-shot translation. We follow Philip et al. (2020) to experiment on TED talks (Qi et al., 2018) and restrict top 20 languages. We also experiment on OPUS-100 (Zhang et al., 2020) to exhaustively explore the capacity of our proposed method in the large-scale dataset. English is the central language of those cases.

To show the strict dependence of non-centered languages, we design two different cases without central language, namely all languages in the set are non-centered. We extract and reorganize Europarl v7 (Koehn et al., 2005) from MMCR4NLP: 1) Triangle case, where each language appears at the target and source sides only once. Our motivation is to build the strict dependence under the non-centered data condition, and each language pair has more training data than IWSLT. Figure 2a shows its translation directions we designed. 2) Square case, that is designed for avoiding strict dependence as indicated in Figure 2b. Our motivation is to avoid completely interconnecting all languages (Fan et al., 2020; Freitag and Firat, 2020) while building balanced data conditions.

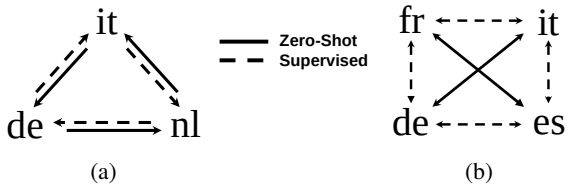


Figure 2: Illustration supervised and zero-shot directions for Triangle and Square cases.

We list details of datasets in Appendix B. And all cases are evaluated via official test sets.

4.2 Experimental Setup

We employ Fairseq (Ott et al., 2019), the open-source implementation, of Transformer (Vaswani et al., 2017) as backbone. Generally, we apply the Moses tokenizer[4] for tokenization and detokenization, and use SentencePiece (Kudo and Richardson, 2018) to learn subword vocabulary. Although the detail of training subword vocabulary for each case has differences, we always train the joint vocabulary including the source and target side, and set *share-all-embedding* in Fairseq. To prevent the unbalanced data size of English-centered datasets from training subword vocabulary, the SentencePiece model is trained by data aggregated from monolingual resources rather than paired resources. We use Adam (Kingma and Ba, 2017) optimizer with the inverse square root schedule in all cases and set different learning rates for different datasets. For fair comparisons, we not only reproduce the Transformer (Vaswani et al., 2017) as Baseline but also reproduce the work of Liu et al. (2021) in all cases, which is denoted by Residual. We always adopt the same hyperparameters setting to prior works and train corresponding subword vocabulary via details described by these works. Specifically, as for the settings for cases of IWSLT, Triangle and Square, we follow the setting of Liu et al. (2021); as for models trained by TED talks and OPUS-100, we follow the setting of Philip et al. (2020) and Zhang et al. (2020), respectively.

We experimented with IWSLT, Triangle, and Square five runs with different random seeds [1,2,3,4,5], to compute the variance for verifying the instability caused by parameters, and other ex-

[4] https://github.com/moses-smt/mosesdecoder

Supervised:	IWSLT		Triangle	Square	TED		OPUS-100	
Method	en→	→en	sup.	sup.	en→	→en	en→	→en
Baseline	31.51	32.93	25.75	32.04	24.23	28.92	19.50	27.60
Philip et al. (2020)	-	-	-	-	24.85	**31.21**	-	-
Zhang et al. (2020)	-	-	-	-	-	-	21.39	27.50
Residual	31.24	32.65	26.25	31.85	22.80	28.19	20.38	26.67
SD	31.63	32.51	26.50	31.97	23.94	28.33	23.60	28.01
FCLL	**31.76**	**33.00**	**26.91**	**32.14**	**25.32**	28.13	**26.17**	**29.33**

Table 2: Averaged BLEU scores on supervised directions. en→ denotes translating from en (l_c) to $\mathbb{L}'$ and →en denotes translating to en from $\mathbb{L}'$; sup. indicates supervised directions in non-centered cases. *Residual* follows Liu et al. (2021) to modify residual connection.

Zero-Shot:	IWSLT		Triangle		Square		TED		OPUS-100			
Method	Z.S.	O.R.	Z.S.	O.R.	Z.S.	O.R.	Z.S.	O.R.	Z.S.	O.R.	F.T.	O.R.
Baseline	16.97	13.95	1.97	93.68	31.18	0.74	10.66	4.16	3.97	63.96	10.11	13.92
Philip et al. (2020)	-	-	-	-	-	-	12.94	-	-	-	-	-
Zhang et al. (2020)	-	-	-	-	-	-	-	-	4.02	54.57	11.98	-
Residual	20.37	1.80	16.60	4.95	30.30	0.77	12.54	3.85	5.14	38.54	11.38	18.30
SD	**21.35**	2.03	19.07	0.92	31.26	0.75	13.03	3.94	4.87	44.07	12.95	12.54
FCLL	21.15	2.05	**20.56**	0.13	**31.49**	0.74	**14.14**	3.74	**6.31**	34.46	**13.65**	11.09

Table 3: Averaged BLEU scores on zero-shot directions. Z.S. column indicates results of zero-shot translation; O.R. denotes the off-target ratio measured by %; F.T. indicates results after fine-tuning, we follow Zhang et al. (2020) to fine-tune 6 languages existing in zero-shot testing.

periments are trained with seed 1. To evaluate results of all experiments, we translate the official test set with beam size 4, and evaluate the translation results by sacreBLEU (Papineni et al., 2002; Post, 2018). We also employ the langdetect[5], which can identify the language of one sentence, to count the off-Target ratio, namely how many sentences are not translated to the correct language. We list detailed experimental settings in Appendix A.

4.3 Results

As described in Table 2, our proposed methods achieve small improvements measured by averaged BLEU scores on supervised directions of IWSLT (+0.25/+0.07), Triangle (+1.16), Square (+0.1), TED (+1.09/-0.79), and OPUS-100 (+6.67/+1.73) compared to Baseline. Liu et al. (2021) speculated that the basic Transformer would overfit more on the supervised direction, and the improvement of zero-shot could hurt supervised translation (Gu et al., 2019; Zhang et al., 2020; Liu et al., 2021). The performance of Residual (Liu et al., 2021) degenerated in TED (-1.43/-0.73) and OPUS-100 (+0.88/-0.93), since the model weakened LS information by trading the generalization ability for zero-shot translation. However, our proposed methods benefit from the additional improvement of decoding the target language by LS modeling (Sachan and Neubig, 2018; Philip et al., 2020). This improvement can counteract the insufficiency of tying artificial language tokens to instruct translation (Arivazhagan et al., 2019). The results of SD can empirically prove the positive impact of CLL, since the performance of SD, which only constructs one CLL, is always between FCLL and Residual on supervised directions. Moreover, the performance of FCLL shows a marked difference (+1.09/-3.08) from the work of Philip et al. (2020). We speculate that the reason is lacking LS structure of l_c and benefiting from the mixture of shared and LS information in CLL. This hypothesis can explain the stable improvements of CLL on Triangle (+1.16) and Square (+0.1) where no l_c exited in training data. We conduct ablation experiments to show the mechanism of CLL in Section 5.4.

Table 3 demonstrates that our methods always give the best scores on zero-shot translations in our experiments. Based on the gain of zero-shot and gain of en→ (Table 2), CLL always positively impacts non-centered languages. In IWSLT, SD performs better than FCLL (+0.2) and performs near FCLL in other cases. It indicates that stacking LS structures is not always optimal for improv-

[5]The tool is not accurate, so, it is just for observing general tendency. (https://github.com/Mimino666/langdetect)

	Zero-Shot			Supervised		
	Baseline	Residual	SD	Baseline	Residual	SD
(1)	14.31	15.06	**16.55**	20.80	20.17	**21.97**
(2)	15.08	16.45	**17.01**	**24.60**	24.38	24.24

Table 4: Averaged BLEU scores of integrating de. Row (1) and Row (2) shows results in $\mathbb{L}_{iwslt} \rightarrow$ de and de $\rightarrow \mathbb{L}_{iwslt}$, respectively.

ing zero-shot translation. It also proves combining tweaking encoding information and improving decoding information would be effective for zero-shot translation. In Triangle, our methods perform stably in the extreme data condition where Baseline totally failed. In Square, all cases have similar performances since these languages do not have strict dependence. Results in TED and OPUS-100 show that our methods also run well in the large-scale dataset. Moreover, we follow Zhang et al. (2020) to fine tune the model by back-translation (Gu et al., 2019) for 6 languages of zero-shot testing, and FCLL achieves a gain of +3.54 BLEU scores to Baseline. These two points show the proposed CLL is orthogonal with other methods excluding LS modeling.

We further noticed that, in Table 2, the performances of all models on zero-shot directions in Square are comparable with each other, and our methods performed stably in Triangle where Baseline is totally failed. The stability of Square case shows the key to improving zero-shot translation is not only large training data (Mattoni et al., 2017), but also the balance of training (Shao and Feng, 2022). The results of Triangle prove CLL is stable in zero-shot translation since it would not be influenced by different data conditions. This feature ensured the effective utilization of shared information. This feature can be proved by the value of off-target rate in Table 3. Given the cost of establishing consistent semantic representation in shared information, confusion about different linguistic features is an inevitable result because the shared information container leads to coupling supervised translation pairs both in theory and practice, however, our proposed methods are always at a relatively lower rate.

4.4 Integrating a new language by few data

The ability to integrate a new language by few data is crucial for low-resource languages when extending a trained MNMT model. To verify this ability of CLL, we fine tune trained SD in IWSLT and extend it to German (de) using bilingual language pairs (en $\leftrightarrow$ de) with 15K sentences per direction, we also fine tune Baseline and Residual as comparison. We follow Liu et al. (2021) to set hyperparameters and update subword vocabulary, as described in Appendix A. Table 4 shows that SD performs better on zero-shot translation, which indicates CLL contributes the cross-lingual knowledge transfer, which indicates that our method is flexible in incorporating low-resource languages.

	IWSLT		Triangle		Square	
	sup.	zero.	sup.	zero.	sup.	zero.
Baseline	0.021	5.280	0.220	0.210	**0.001**	0.016
Residual	0.067	0.270	**0.004**	0.900	0.003	0.055
SD	**0.012**	**0.051**	0.018	0.900	0.002	**0.001**
FCLL	0.025	0.074	0.018	**0.140**	0.004	0.014

Table 5: Variance computed from averaged BLEU scores among five runs in IWSLT, Triangle, and Square with different random seeds. sup. and zero. indicate supervised translation and zero-shot translation, respectively. Smaller variance means a more stable result.

5 Discussion and Analyses

5.1 Instability of Zero-Shot Translation

In this paper, we describe the instability from two related perspectives: 1) Instability of training; 2) Instability of data conditions. For the first point, Table 5 shows the variance for different models counted from five experiments with different seeds for initialization. The small value of variance on supervised translation among the four models shows that supervised training always is a relatively stable process. However, the training process of zero-shot translation is sensitive to initial parameters (Gu et al., 2019), since the variance of zero-shot translation is always higher than the variance of supervised translation. Our methods always achieved the lowest variance on zero-shot translation. For the second point, Table 3 shows that Baseline has completely lost its ability of zero-shot translation in Triangle, although the amount of training samples of Triangle is relatively higher than IWSLT that can result in a good performance on zero-shot translation. On the other hand, Square performs excellently on zero-shot translation and its performance is even closing to the performance of supervised translation (Table 3), although it is non-centered data condition as same as Triangle and it is not completely interconnecting all languages (Fan et al., 2020; Freitag and Firat, 2020). These comparisons proved that the data condition impacts the learning of zero-shot translation.

	Condition	it→nl	ro→it	nl→ro
(1)	Baseline	18.69	16.43	14.13
(2)	(1)+additional pairs	23.27	22.17	21.61
(3)	(2)+reduce data	22.35	21.31	20.96
	SD	22.13	20.27	20.42

Table 6: Variation of different conditions in IWSLT. SD is the performance under original setting.

We further notice that Baseline has a high variance in IWSLT yet a small variance in Triangle. We speculate that the strict dependence of non-centered languages caused instability, and the degree of dependence influences the expression of instability. Specifically, Baseline tends to build cross-linguistic representations in IWSLT, but the strict dependence would couple representations of non-centered languages to the central language to lead to a high off-target ratio in testing zero-shot translation (Table 3). And the higher variance of Baseline in IWSLT means that the model may find a special set of initial parameters to escape from the negative influence of strict dependence. Moreover, the small variance of Baseline in Triangle means that the model completely cannot find a special set among the five times experiments, since Triangle has the most severe dependence of non-centered languages.

To prove our speculation, we create two artificial setups based on IWSLT to re-train Baseline and show results measured by BLEU in Table 6. Specifically, for Row (2), we append three language pairs (it → ro, ro → nl, nl → it) with 30k sentences per pair to balance the dependence (Rios et al., 2020); for Row (3), we sample a random subset of 90K sentences from training data of IWSLT of 145K sentences per translation direction and we append additional pairs as (2). These substantial gains (up to +7.48) of Row (2) in Table 6 proved our viewpoint that data conditions impact the performance of zero-shot translation. Once the model disentangled the strict dependence by appending additional pairs, the model would achieve considerable gains (up to +6.83), although the training samples have been reduced to be smaller than the original setting shown by Row (1). Moreover, the performance of SD is comparable to these artificial cases.

So far we can conclude that the strict dependence of non-centered languages closely influences the zero-shot translation. And our motivation for disentangling the dependence by improving the weak representations of non-centered languages is effective. We will discuss the mechanism of CLL in Section 5.4.

Method	Supervised	Zero-Shot	Off(%)
Residual	32.57	20.74	1.67
FCLL	32.95	21.00	1.52
SD	32.48	21.16	1.97
Residual w/o t	23.72	0.56	-
FCLL w/o t	32.88	20.85	1.35
SD w/o t	32.62	19.46	2.10

Table 7: Averaged BLEU scores of models training without language tokens (w/o t) in IWSLT.

	it→		ro→		nl→	
	ro	nl	it	nl	it	ro
FCLL	21.14	21.92	20.43	22.12	19.45	20.98
Omitted	-0.16	-0.08	-0.37	-0.21	-0.87	-0.74
SD	20.88	22.13	20.27	22.75	20.42	20.51
Omitted	-1.79	-1.40	-2.13	-1.68	-2.41	-2.59

Table 8: Variation of BLEU scores after omitting language tokens in testing.

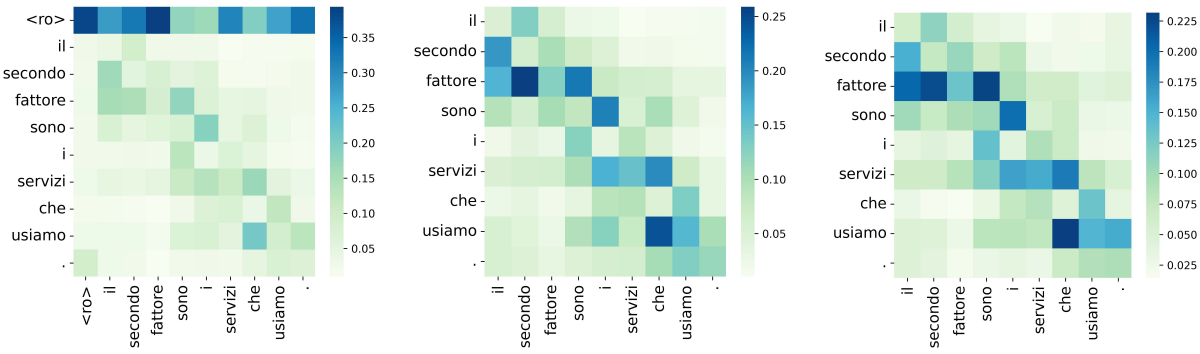
5.2 Translation Instructions

We re-train our models in IWSLT without language tokens (Johnson et al., 2017), and Table 7 shows the result. First, slight performance gains were observed on supervised directions in FCLL and SD. Figure 3 is a heat map showing self-attention weights of FCLL with and without language tokens. Figure 3a shows one possibility is artificial language tokens (Johnson et al., 2017) might disturb the semantic representation for actual words, since the language token <ro> dominated in self-attention weights. Figure 3b shows the distribution of actual words weights by training without language tokens. Figure 3c presents the attention weights when omitting <ro> in testing the model trained with language tokens, and we observed a similar tendency with the plot in Figure 3b.

Second, Table 7 shows our methods stably maintain cross-linguistic representation although no language tokens were inserted to instruct translation directions both for supervised and zero-shot directions. On the contrary, other methods completely lost their ability of zero-shot translation. These analyses indicate that CLL has a strong capability to instruct multilingual translation.

5.3 Full Layers vs. Single Layer

To verify whether the number of CLL affects the performance of MNMT model, we translate the test set omitting artificial language tokens by trained



(a) CLL (b) CLL w/o token (c) CLL omitted token

Figure 3: Maps of Self-Attention in which translating one sentence of ro → it.

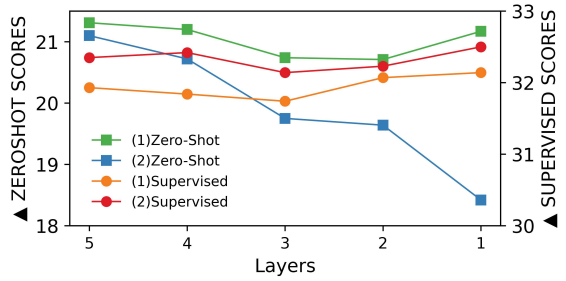


Figure 4: Variation of BLEU scores in which training on different CLL layers. (1) modified the residual connection as SD; (2) did not modify it.

	supervised			zero-shot		
	it→	nl→	de→	it→	nl→	de→
(1)	26.80	26.28	26.00	18.44	19.80	16.36
(2)	25.19	25.77	25.56	0.64	0.75	1.02
(3)	24.85	24.46	25.49	_	_	_

Table 9: Averaged BLEU scores of ablation study. Row (1) shows results in Triangle; Row (2) shows results after ablation; Row (3) means to calculate scores of zero-shot translation by treating the supervised translation results as reference data.

it→de	Ablated LSL of de from CLL
Input:	<de> la quarta priorità concerne l'attenzione che occorre prestare ai nuovi rischi.
Expected Output:	die vierte priorität gilt den neuen risiken.
Actual Output:	de vierde prioriteit is de aandacht die moet worden besteed aan nieuwe risico 's.

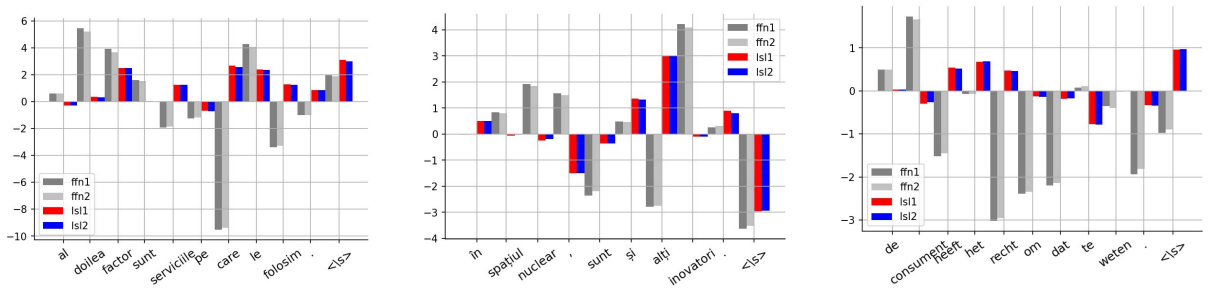
Table 10: Ablated testing SD trained in Triangle. The output of the model rolls back to nl (Dutch, the supervised direction).

FCLL and SD in IWSLT. Table 8 demonstrates the model with more CLL layers has stronger robustness since scores of FCLL degenerated less than SD. To further investigate the effect of the number of CLL layers, we re-train Transformer models with different numbers of CLL layers based on IWSLT in two cases. Specifically, in case (1), we modify the residual connection of models as same as the operation of SD, but we do not modify any architecture in encoder of Transformer in case (2). Then, we follow the idea of Liu et al. (2021) to remove the CLL layers in the decoder from the top-most and bottom-most positions until the configuration in which only a single CLL layer is preserved in the middle-position decoder among all decoders of these models.

Figure 4 shows that the zero-shot performance of models in (2) degenerated with the reduction of the number of CLL layers, although the supervised performance always kept in the same magnitude. It proves that the increase in the number of CLL layers has a positive impact on the zero-shot translation. However, almost no clear variations were observed in (1) of Figure 4. One possibility of the lower supervised performance of (1) when compared with (2) in Figure 4 is the weaken language specific information in the encoder by removing the residual connection (Liu et al., 2021). Likewise, the zero-shot performance of (1) is not sensitive to the variation of the number of CLL layers since weakening the capacity of the encoder could partially offset CLL's gains in decoder. We conclude that the architecture of SD is relatively-optimal in small-scale dataset because it is lightweight yet comparable with FCLL, and FCLL is more stable where data condition is complex or large.

5.4 Disentangling Coupled Representation

Ablation Study To investigate the significance of CLL, we ablate LSL from CLL of trained SD



(a) IWSLT:it → ro (b) IWSLT:en → ro (c) Triangle:de → nl

Figure 5: Visualization of layer attributions. ffn indicates FFN, lsl means LSLs, 1 or 2 means it is 1st or 2nd fully connected neural network of this component. A higher absolute value indicates more contribution for result.

in Triangle, namely only use FFN, to re-translate the test set. Row (2) of Table 9 show the degeneration of supervised translation is not significant, but completely losing zero-shot translation capability. However, we observed that the zero-shot translation rollbacks to supervised directions after ablation via analyzing failure cases. As shown in Table 10, the zero-shot translation of it → de will be biased to it → nl due to ablating the layer of de[6]. Thus, we calculated the BLEU scores of zero-shot translation by treating the test set of it → nl as reference data in testing Row (3) of Table 9. The slight degeneration of Row (3) strongly proved that FFN has built a consistent semantic representation which has been coupled to supervised directions.

Layer Attributions The layer attribution[7] can quantify the contributions of one component by integrated gradients (Sundararajan et al., 2017). We designed 3 scenarios to observe these attributions in details: a) The zero-shot translation based on centered case; b) The supervised translation based on centered case; c) The zero-shot translation based on non-centered case. Figure 5 demonstrates: 1) FFN always plays the main role in translation; 2) Generally, the contributions of CLL are on the contrary of FFN in LS words, but they have similar contributions in common words, especially the punctuation.

These results proved our viewpoint in Section 5.1 again. Specifically, the shared representations built in FFN potentially enable cross-linguistic transferring, but the strict dependence of non-centered languages would hamper freely transferring since cross-linguistic information is coupled with supervised translation directions. Therefore, the significance of LSL in CLL practically is to provide independent LS information to disentangle the coupled representation, namely counteract the negative influence of the dependence, to present a correct LS representation in decoding.

6 Conclusion

In this work, we supplement the theory of zero-shot translation with the strict dependence of non-centered languages, and we describe the instability of zero-shot translation. To counteract the influence of the dependence, we proposed a simple yet effective method that employs LS modeling by adapting to non-centered languages. Our analysis based on layer attribution demonstrated that LS information is conducive to disentangling the coupled model representation. Our experiments on various datasets and different data conditions show that our proposed method outperforms in performance and complexity.

7 Ethical Considerations

The potential ethical risk of our work is the usage of multilingual datasets including IWSLT, Europarl, TED talks and OPUS-100, since these datasets might contain social biases, especially in the Europarl, in which predominant European languages might constitute stereotypes. Those biases would be represented in the trained model and could be amplified by integrating one new language out of trained language families since no special treatment is performed to mitigate the biases. Generally, this method can be landed in the industry under sufficient anti-prejudice measures.

Acknowledgements

This work was in part supported by JSPS KAKENHI Grant Numbers 21H05054.

[6] we report more examples in Appendix D in which including long and short sentences in different cases.

[7] We employ Captum (https://github.com/pytorch/captum) for computing attributions.

References

Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Roee Aharoni, Melvin Johnson, and Wolfgang Macherey. 2019. The missing ingredient in zero-shot neural machine translation.

Ankur Bapna and Orhan Firat. 2019. Simple, scalable adaptation for neural machine translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1538–1548, Hong Kong, China. Association for Computational Linguistics.

Graeme Blackwood, Miguel Ballesteros, and Todd Ward. 2018. Multilingual neural machine translation with task-specific attention. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3112–3122, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Mauro Cettolo, Marcello Federico, Luisa Bentivogli, Niehues Jan, Stüker Sebastian, Sudoh Katsuitho, Yoshino Koichiro, and Federmann Christian. 2017. Overview of the iwslt 2017 evaluation campaign. In *International Workshop on Spoken Language Translation*, pages 2–14.

Yong Cheng, Ankur Bapna, Orhan Firat, Yuan Cao, Pidong Wang, and Wolfgang Macherey. 2022. Multilingual mix: Example interpolation improves multilingual neural machine translation.

Raj Dabre and Sadao Kurohashi. 2019. Mmcr4nlp: Multilingual multiway corpora repository for natural language processing.

Kedar Dhamdhere, Mukund Sundararajan, and Qiqi Yan. 2019. How important is a neuron. In *International Conference on Learning Representations*.

Carlos Escolano, Marta R. Costa-jussà, José A. R. Fonollosa, and Mikel Artetxe. 2021. Multilingual machine translation: Closing the gap between shared and language-specific encoder-decoders. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 944–948, Online. Association for Computational Linguistics.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, Naman Goyal, Tom Birch, Vitaliy Liptchinsky, Sergey Edunov, Edouard Grave, Michael Auli, and Armand Joulin. 2020. Beyond english-centric multilingual machine translation.

Orhan Firat, Kyunghyun Cho, Baskaran Sankaran, Fatos T. Yarman Vural, and Yoshua Bengio. 2017. Multi-way, multilingual neural machine translation. *Computer Speech & Language*, 45:236–252.

Markus Freitag and Orhan Firat. 2020. Complete multilingual neural machine translation. In *Proceedings of the Fifth Conference on Machine Translation*, pages 550–560, Online. Association for Computational Linguistics.

Jiatao Gu, Yong Wang, Kyunghyun Cho, and Victor O.K. Li. 2019. Improved zero-shot neural machine translation via ignoring spurious correlations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1258–1268, Florence, Italy. Association for Computational Linguistics.

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

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A method for stochastic optimization.

Philipp Koehn et al. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pages 79–86. Citeseer.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Surafel M. Lakew, Marcello Federico, Matteo Negri, and Marco Turchi. 2019. Multilingual neural machine translation for zero-resource languages.

Jason Lee, Kyunghyun Cho, and Thomas Hofmann. 2017. Fully Character-Level Neural Machine Translation without Explicit Segmentation. *Transactions of the Association for Computational Linguistics*, 5:365–378.

Zehui Lin, Liwei Wu, Mingxuan Wang, and Lei Li. 2021. Learning language specific sub-network for multilingual machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 293–305, Online. Association for Computational Linguistics.

Danni Liu, Jan Niehues, James Cross, Francisco Guzmán, and Xian Li. 2021. Improving zero-shot translation by disentangling positional information. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1259–1273, Online. Association for Computational Linguistics.

Yichao Lu, Phillip Keung, Faisal Ladhak, Vikas Bhardwaj, Shaonan Zhang, and Jason Sun. 2018. A neural interlingua for multilingual machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 84–92, Brussels, Belgium. Association for Computational Linguistics.

Giulia Mattoni, Pat Nagle, Carlos Collantes, and Dimitar Shterionov. 2017. Zero-shot translation for indian languages with sparse data. In *Proceedings of the MT Summit*, pages 1–10.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.

Myle Ott, Sergey Edunov, David Grangier, and Michael Auli. 2018. Scaling neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 1–9, Brussels, Belgium. Association for Computational Linguistics.

Xiao Pan, Mingxuan Wang, Liwei Wu, and Lei Li. 2021. Contrastive learning for many-to-many multilingual neural machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 244–258, Online. Association for Computational Linguistics.

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

Ngoc-Quan Pham, Jan Niehues, Thanh-Le Ha, and Alexander Waibel. 2019. Improving zero-shot translation with language-independent constraints. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 13–23, Florence, Italy. 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, Online. Association for Computational Linguistics.

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.

Ye Qi, Devendra Sachan, Matthieu Felix, Sarguna Padmanabhan, and Graham Neubig. 2018. When and why are pre-trained word embeddings useful for neural machine translation? In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 529–535, New Orleans, Louisiana. Association for Computational Linguistics.

Annette Rios, Mathias Müller, and Rico Sennrich. 2020. Subword segmentation and a single bridge language affect zero-shot neural machine translation. In *Proceedings of the Fifth Conference on Machine Translation*, pages 528–537, Online. Association for Computational Linguistics.

Devendra Sachan and Graham Neubig. 2018. Parameter sharing methods for multilingual self-attentional translation models. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 261–271, Brussels, Belgium. Association for Computational Linguistics.

Chenze Shao and Yang Feng. 2022. Overcoming catastrophic forgetting beyond continual learning: Balanced training for neural machine translation.

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.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 3319–3328. JMLR.org.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Qian Wang and Jiajun Zhang. 2021. Parameter differentiation based multilingual neural machine translation.

Yining Wang, Jiajun Zhang, Feifei Zhai, Jingfang Xu, and Chengqing Zong. 2018. Three strategies to improve one-to-many multilingual translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2955–2960, Brussels, Belgium. Association for Computational Linguistics.

Liwei Wu, Shanbo Cheng, Mingxuan Wang, and Lei Li. 2021. Language tags matter for zero-shot neural machine translation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3001–3007, Online. Association for Computational Linguistics.

Yilin Yang, Akiko Eriguchi, Alexandre Muzio, Prasad Tadepalli, Stefan Lee, and Hany Hassan. 2021. Improving multilingual translation by representation

and gradient regularization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7266–7279.

Poorya Zaremoodi, Wray Buntine, and Gholamreza Haffari. 2018. Adaptive knowledge sharing in multi-task learning: Improving low-resource neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 656–661, Melbourne, Australia. 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. In *International Conference on Learning Representations*.

Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. 2020. Improving massively multilingual neural machine translation and zero-shot translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1628–1639, Online. Association for Computational Linguistics.

A Detailed Settings

IWSLT & Triangle & Square We follow Liu et al. (2021) to set 5 encoder/decoder layers with 8 attention heads, embedding size of 512, inner size of 2048, dropout rate of 0.3, dropout rate of CLL layer of 0.3, maximum learning rate of 0.0005 and label smoothing rate of 0.1. However, we decrease dropout rate to 0.1 and dropout rate of CLL layer to 0.2 in Square that is a bigger case than others. The size of subword vocabulary is 40K for each case. In training, we set the maximum batch size per GPU to 4,000 tokens, and train on 4 GPUs. We train for 100K steps for IWSLT and Triangle, but train for 500K steps for Square. We sample the supervised and zero-shot translation directions from the dev set of MMCR4NLP as the validation dataset in training.

TED talks We follow Philip et al. (2020) to set 6 encoder/decoder layers with 4 attention heads, embedding size of 512, inner size of 1024, dropout rate of CLL layer of 0.3, maximum learning rate of 0.0005 and label smoothing rate of 0.1. However, we set the dropout rate to 0.2 to get better performances. The size of subword vocabulary is 70K. In training, we set the maximum batch size per GPU to 4,000 tokens, and train on 4 GPUs. We train for 90 epochs to ensure models convergent. We only sample dev sets of supervised directions translating as the validation dataset in training. We also follow Philip et al. (2020) to use mixed-precision (Ott et al., 2018) in training.

OPUS-100 We follow Zhang et al. (2020) to set 6 encoder/decoder layers with 8 attention heads, embedding size of 512, inner size of 2048, dropout rate to 0.1, dropout rate of CLL layer of 0.2, maximum learning rate of 0.0007 and label smoothing rate of 0.1. We directly reuse their published subword vocabulary[8]. In training, we set the maximum batch size per GPU to 6,000 tokens, and train on 8 GPUs[9] for 500K steps. We follow Zhang et al. (2020) to sample top 200 sentences in dev sets of supervised directions translating as the validation dataset in training.

In fine-tuning, we follow Zhang et al. (2020) to back-translate the training resource to get the pseudo resource, then we merge real and pseudo resources to train 4 epochs, and we update the pseudo training resource after each epoch in training. We set 500 warm-up steps at the beginning of fine-tuning, reset the optimizer, and training with maximum learning rate of 0.0003.

Integrating de in IWSLT Based on the trained model in IWSLT, we learn a new SentencePiece model with 10K vocabulary size to acquire a dictionary for de. Then we append the new dictionary to the end of the previously learned dictionary of IWSLT, meanwhile, we keep the order of the previous part unchanged. Due to the increased number of unique tokens, we resize token embedding and initialize new vectors as the average of existing embedding perturbed by random noise. When fine-tuning, we set the learning rate as the value at the end of the previous training, freeze parameters of CLL layers of existing languages, initialize parameters of CLL layers for de by averaging existing CLL layers, and include the original training data of IWSLT to prevent the shared information from tending to translate de.

[8] https://github.com/bzhangGo/zero

[9] We use Fairseq command line of *–update-freq 2* to simulate the efficiency of 8 GPUs by 4 GPUs.

B Dataset Details

Dataset case	Languages	# zero-shot directions	# sent. per direction
IWSLT	{en, it, ro, nl}	6	145K
Europarl Triangle	{__, it, nl, de}	3	200K
Europarl Square	{__, fr, it, de, es}	4	1M
TED	{ en, ar, bg, de, es, fa, fr, he, hu, it, ja, ko, nl, pl, pt-br, ro, ru, tr, vi, zh-cn }	342	140K ~210K
OPUS-100	{ en, an, as, be, bg, bn, br, bs, ca, cs, cy, da, de, el, es, fa, fr, fy, ga, gd, gl, gu, hi, hr, hy, is, it, ku, li, lt, lv, mk, mr, nb, ne, nl, nn, no, oc, or, pa, pl, ps, pt, ro, ru, sh, si, sk, sl, sq, sr, sv, tg, uk, ur, wa, yi, az, kk, ky, tk, tr, tt, ug, uz, dz, my, zh, et, fi, hu, se, id, km, mg, ms, vi, ig, rw, xh, yo, zu, kn, ml, ta, te, eo, eu, ja, ko, ka, mn, th}	30	2K ~1M

Table 11: Overview of datasets. The underline denotes the l_c, and the underline with blank represents non-centered condition, i.e. no English.

C Distribution of Language-Specific Information Weights

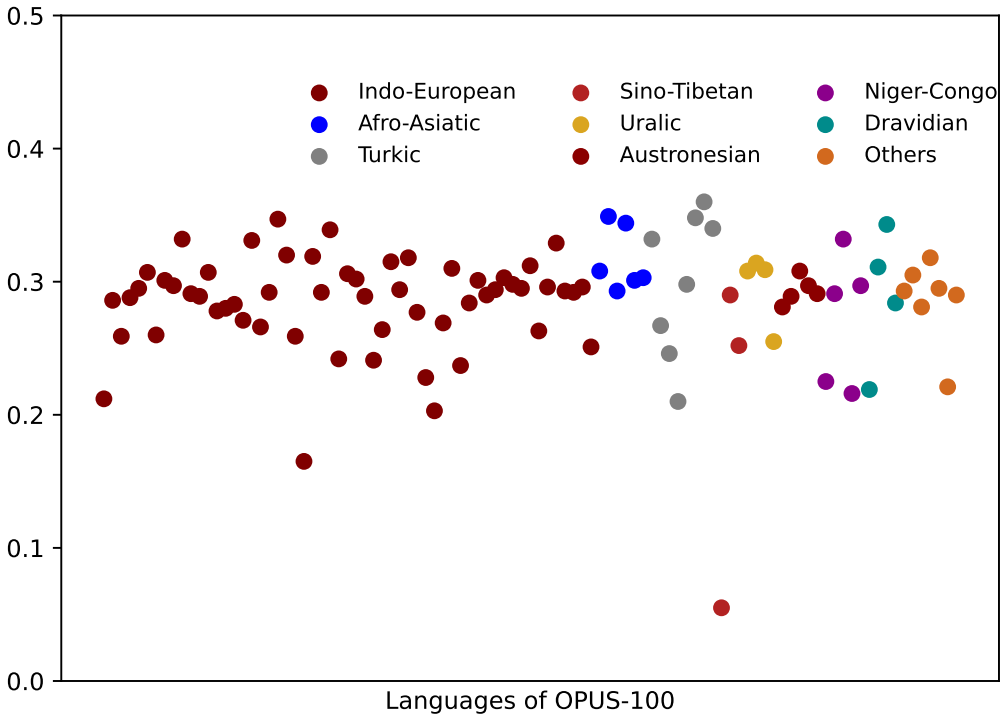


Figure 6: Averaged weights over all layers in FCLL model. The x-coordinate is sorted by languages showed in Table 11. For languages with the same amount of training resources, languages from the same language family have relatively similar weights.

D Translation Examples of Ablation

Supervised: it→nl	Ablated LSL of nl from CLL	Language
Input:	<nl> parlo adesso per esperienza personale: da anni nell'industria dell'aviazione civile e con la commissione siamo infarciti di deregolamentazione, eppure,in relazione ai diritti aeroportuali, ci viene detto adesso che la risposta è la regolamentazione.	it
Expected Output:	ik spreek nu namens mijzelf: al vele jaren wordt ons nu binnen de burgerluchtvaartindustrie en met de commissie een dieet voorgeschoteld van deregulatie en toch, waar het gaat om luchthavenbelasting, wordt ons nu verteld dat regelgeving het antwoord is.	nl
Actual Output:	ik spreek nu uit persoonlijke ervaring: al jaren in de burgerluchtvaartindustrie en met de commissie zijn we gedwongen tot deregulering, maar wat de luchthavengelden betreft, wordt ons nu gezegd dat het antwoord de regelgeving is.	nl
Input:	<nl> tutte le cose importanti sono già state dette.	it
Expected Output:	al het belangrijke is reeds gezegd.	nl
Actual Output:	al het belangrijke is reeds gezegd.	nl
Supervised: de→it	Ablated LSL of it from CLL	Language
Input:	<it> der verbraucher hat ein recht darauf, das zu wissen.	de
Expected Output:	il consumatore ha il diritto di saperlo.	it
Actual Output:	il consumatore ha il diritto di saperlo.	it
Zero-Shot: it→de	Ablated LSL of de from CLL	Language
Input:	<de> il recepimento di parte dell'acquis nel primo pilastro apre la strada alla comunitarizzazione di questa politica e consente di adottare anche rimedi in relazione alla nebulosa schengen, come amava chiamarla il mio predecessore.	it
Expected Output:	mit der teilweisen übernahme des acquis in den ersten pfeiler stehen uns nun alle wege offen, diese politik zu vergemeinschaften und licht in la nébuleuse schengen zu bringen, wie es mein vorredner beschrieb.	de
Actual Output:	de omzetting van een deel van het acquis in de eerste pijler maakt de weg vrij voor de communautarisering van dit beleid en maakt het mogelijk dat er ook oplossingen worden gevonden voor de nebulosa schengen, zoals mijn voorganger zei.	nl
Input:	<de> la quarta priorità concerne l'attenzione che occorre prestare ai nuovi rischi.	it
Expected Output:	die vierte priorität gilt den neuen risiken.	de
Actual Output:	de vierde prioriteit is de aandacht die moet worden besteed aan nieuwe risico 's.	nl
Zero-Shot: nl→it	Ablated LSL of it from CLL	Language
Input:	<it> die solidariteit en die noodzaak tot samenwerking geldt ook als zich in de toekomst problemen voordoen, bijvoorbeeld bij interne migratiestromen.	nl
Expected Output:	questa sicurezza e la necessità di una collaborazione sono essi stessi potenziali problemi futuri, ad esempio per quanto riguarda la migrazione interna.	it
Actual Output:	diese solidarität und die notwendigkeit der zusammenarbeit gelten auch in zukunft, z. b. in bezug auf die migrationsströme.	de

Table 12: Some examples of translation by trained SD in the Triangle, in which ablating LS layers from CLL. The long sentence of supervised translation has degeneration compared with short sentences but is kept in the correct direction. These zero-shot translations are biased to supervised directions.