Languages Transferred Within the Encoder: On Representation Transfer in Zero-Shot Multilingual Translation

Anonymous ACL submission

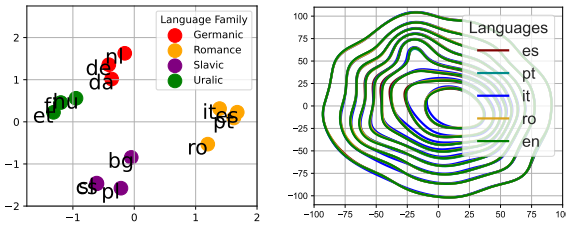
Abstract

Understanding representation transfer in multilingual neural machine translation (MNMT) can reveal the reason for the zero-shot translation deficiency. In this work, we systematically analyze the representational issue of MNMT models. We first introduce the identity pair, translating a sentence to itself, to address the lack of the base measure in multilingual investigations, as the identity pair can reflect the representation of a language within the model. Then, we demonstrate that the encoder transfers the source language to the representational subspace of the target language instead of the language-agnostic state. Thus, the zero-shot translation deficiency arises because the representation of a translation is entangled with other languages and not transferred to the target language effectively. Based on our findings, we propose two methods: 1) low-rank language-specific embedding at the encoder, and 2) language-specific contrastive learning of the representation at the decoder. The experimental results on Europarl-15, TED-19, and OPUS-100 datasets show that our methods substantially enhance the performance of zero-shot translations by improving language transfer capacity, thereby providing practical evidence to support our conclusions.[1]

1 Introduction

State-of-the-art neural machine translation systems are adaptable to multilingualism, resulting in a single encoder-decoder model that executes arbitrary translations by adding a tag specified to the target language at the beginning of source sentence (Firat et al., 2016; Johnson et al., 2017; Wu et al., 2021). Multilingual neural machine translation (MNMT) is theoretically attractive because zero-shot translations, i.e., translations unseen in training, allow the training of a multilingual model with minimal cost.

[1]We will release all code and analysis scripts on GitHub for reproduction if our paper is accepted.



(a) Cluster by target languages (b) Semantic alignments

Figure 1: Different analytical methods lead to different conclusions. 1a means the target language family clusters the representations of translations from English (en) to other languages through the encoder. 1b indicates the encoder semantically aligns different source languages. Language codes in this work follow ISO 639-1, and Appendix D provides details of those figures.

Unfortunately, the performance of zero-shot translations always lags behind (Aharoni et al., 2019; Arivazhagan et al., 2019a; Gu et al., 2019; Yang et al., 2021; Pan et al., 2021; Chen et al., 2023).

Representational analysis in MNMT models can guide the improvement of zero-shot translation. However, two contrary opinions are demonstrated by prior works: (1) the encoder clusters translation representations based on the target language (Kudugunta et al., 2019; Liu et al., 2021; Tan and Monz, 2023; Stap et al., 2023; Sun et al., 2024), as illustrated in Figure 1a; (2) an ideal encoder is expected to learn language-agnostic representations, capturing general cross-lingual features that are transferable across languages (Pan et al., 2021; Gu and Feng, 2022; Gao et al., 2023; Bu et al., 2024), as shown in Figure 1b. In this work, we aim to analyze and reconcile this discrepancy. We first introduce the identity pair, a pseudo pair translating a sentence to itself. Specifically, the analyses conducted by those prior works rely on real translation pairs, leading to inaccurate results, as a translation pair cannot serve as a base measure for another pair. The identity pair, however, addresses this issue by serving as a proxy for the optimal representation of a language instead of a translation pair. Then, multiple analytical methods are employed to show the

representation transfer within MNMT models. Our findings offer a unified perspective on these two opinions: the encoder transfers translation representations into the target language subspace, where different source languages are semantically aligned. Thus, the zero-shot translation deficiency stems from the failure to transfer the translation representation to the target language, as it becomes entangled with representations of other languages in the encoder.

Guided by our findings, we propose two methods for the encoder and decoder, respectively, to improve multilingual representations: **Lo**w-Rank **L**anguage-specific **E**mbedding (LoLE) is applied to bias the representations in the subspaces of target languages at the encoder; and **L**anguage-specific **C**ontrastive **L**earning of **R**epresentations (LCLR) is applied at the decoder to isolate representational space across languages. We evaluated the proposed methods on three benchmarks, Europarl-15 (Koehn et al., 2005), TED-19 (Ye et al., 2018), and OPUS-100 (Zhang et al., 2020a; Yang et al., 2021), for two automatic metrics, SacreBLEU (Post, 2018) and BERTScore (Zhang et al., 2020b). The experimental results show that our methods outperform strong baselines in training from scratch because of improved representational transferability. Our methods also perform effectively in fine-tuning, even though pre-trained models are trained by different strategies of language tags, which proves that target language information on the encoder side consistently benefits MNMT.

2 Background

2.1 Multilingual Neural Machine Translation

Johnson et al. (2017); Wu et al. (2021) demonstrated that the training strategy of adding a language tag at the beginning of the input sentence on the encoder side boosts the zero-shot translation capacity of the MNMT model. Given a multilingual corpus $\mathbb{C}$ that covers a set of t languages, a set of their corresponding language tags exists: $\mathbb{L} = \{l_1, l_2, \ldots, l_t\}$. For a source-target sentence pair $(\boldsymbol{x}, \boldsymbol{y})$, i.e., $\boldsymbol{x} = x_1, x_2, \ldots, x_n$ and $\boldsymbol{y} = y_1, y_2, \ldots, y_m$, the training data consists of a pair in form of $(\boldsymbol{x}, l, \boldsymbol{y})$, where l is the language tag of $\boldsymbol{y}$ that instructs translation from $\boldsymbol{x}$ into language l. The model is trained over all pairs in $\mathbb{C}$ to optimize the following cross-entropy loss:

$$\mathcal{L}_{ce} = -\sum_{\boldsymbol{x},l,\boldsymbol{y}\in\mathbb{C}} \log p(\boldsymbol{y}|l, \boldsymbol{x}; \theta), \qquad (1)$$

where $p(\boldsymbol{y}|l, \boldsymbol{x}; \theta)$ is the probability distribution of $\boldsymbol{y}$ and θ represents the model parameters.

2.2 The Discrepancy in Prior Works

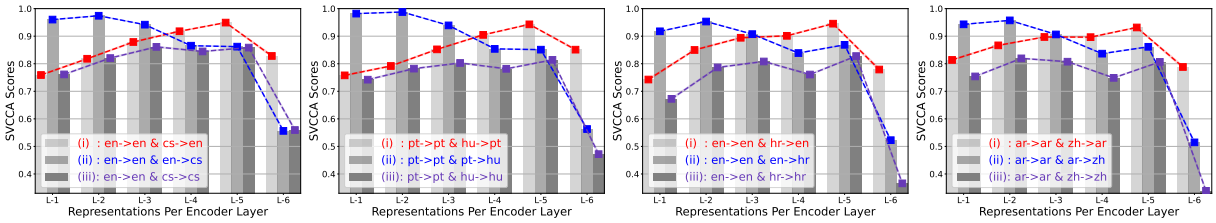
Pan et al. (2021); Gao et al. (2023); Bu et al. (2024) state that, for an encoder-decoder MNMT model, an ideal encoder is regarded as transferring the source sentence into a language-agnostic state, preserving only semantic information. As evidence, the t-distributed stochastic neighbor embedding (t-SNE) (van der Maaten and Hinton, 2008), which can convert similarities between vectors into joint probabilities, has been used to show that representations of sentences from different languages are aligned at the output of the encoder when sharing the same semantics. However, this result contrasts with the findings of Kudugunta et al. (2019); Stap et al. (2023); Tan and Monz (2023). Specifically, using the singular value canonical correlation analysis (SVCCA) (Raghu et al., 2017) to compare the similarity between two vectors, i.e., the sentence representations of two translations, reveals that the encoder tends to transfer the representation into a state with target language features.

We argue that this discrepancy stems from the lack of a base measure. Namely, those works always compare the representations of real translation pairs in which different analysis methods lead to different results. For instance, the translation from English to German, denoted by en→de, cannot be accurately measured by comparing it with another translation from a different language x→de, because en→de is expected to be measured by the language representation of either de or en. Thus, proposing a base measure is necessary to draw the same conclusion from different analysis methods, e.g., t-SNE or SVCCA[2].

3 Investigating Representation Transfer in MNMT

We conduct preliminary experiments to investigate representations by introducing identity pairs as base measures using two different datasets, Europarl-15 (Koehn et al., 2005) and TED-19 (Ye et al., 2018), which are introduced in Appendix B in detail. Then, following Kudugunta et al. (2019), our investigation is based on Transformer models with 6 encoder and decoder layers, where Appendix C introduces the detailed model settings.

[2]We follow Liu et al. (2021) to measure SVCCA scores at the sentence level, which is introduced in Appendix A.



(a) (en, cs) in Europarl. (b) (pt, hu) in Europarl. (c) (en, hr) in TED. (d) (ar, zh) in TED.

Figure 2: Visualizations of layer-wise SVCCA scores for the encoder. (①, ②) indicate the source language and target language, respectively. The analyzed models have 6 encoder layers, and the analysis based on models with 8 and 10 encoder layers shows the same tendency, which is shown in Appendix E.

3.1 Identity Pairs

An identity pair refers to a sentence pair translating from one sentence into itself to represent the optimal state of processing language features, i.e., the semantics and syntax of the source sentence by the model. On the encoder side, we derive the representation from a language translating to itself, i.e., $(\boldsymbol{x}, l', \boldsymbol{x})$, where l' is the language tag of $\boldsymbol{x}$, with the aim of recovering the source sentence from the hidden representations without inference on the decoder side. We also derive the representation in the decoder from the gold translation of $(\boldsymbol{x}, l', \boldsymbol{x})$.

We use SacreBLEU (Papineni et al., 2002; Post, 2018) to evaluate the translation quality of 6 identity pairs, which are generated by inference. The scores of en→en, de→de, and pt→pt in Europarl-15 are 73.49, 61.04, and 71.97, which significantly outperform 44.04 of de→en, 36.63 of en→de, and 46.24 of en→pt, respectively. Similarly, en→en, tr→tr, and vi→vi in TED-19 obtain scores of 72.52, 36.58 and 59.26, which are higher than 34.92 of de→en, 14.81 of en→tr, and 29.78 of en→vi, respectively. Such high scores in the identity pair are caused by that short sentences are recovered from hidden representations perfectly, and long sentences only have a few changes in word selection. Such evidence suggests that identity pairs can serve as base measures for comparing representations because the identity pair is a proxy for the optimal representation of a language, specifically, x→en are expected to be close to en→en in the representational space.

3.2 Language Transfer Within the Encoder

Given two languages ① and ②, two sentences $\boldsymbol{x}$ of ① and $\boldsymbol{y}$ of ② are grouped by three cases when comparing the representations of the encoder: (i) comparing $(\boldsymbol{x}, l^{①}, \boldsymbol{x})$ and $(\boldsymbol{y}, l^{①}, \boldsymbol{x})$ to show how target language features are encoded; (ii) comparing $(\boldsymbol{x}, l^{①}, \boldsymbol{x})$ and $(\boldsymbol{x}, l^{②}, \boldsymbol{y})$ to show how source language features are encoded; (iii) comparing two different identities, $(\boldsymbol{x}, l^{①}, \boldsymbol{x})$ and $(\boldsymbol{y}, l^{②}, \boldsymbol{y})$[3].

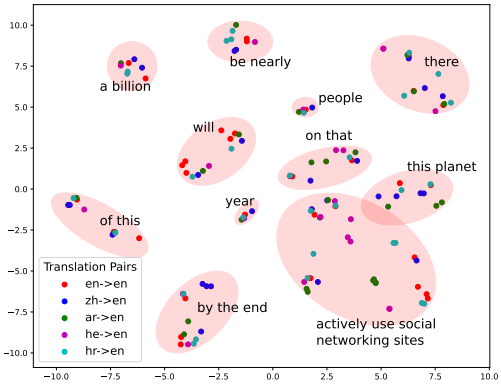
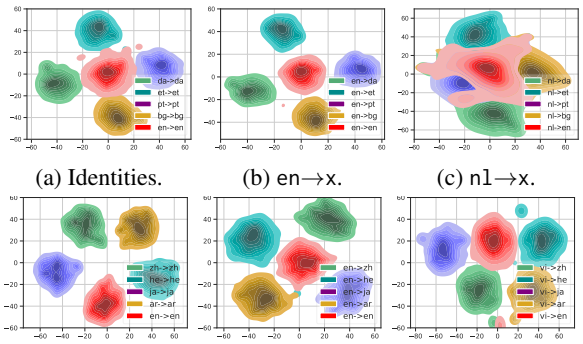


Figure 3: t-SNE plot of the token-level alignment between en→en and x→en in TED-19. Each point is a token's representation collected from the output of the encoder. Representations of different tokens are clustered by the semantics, which are denoted by English phrases, where the overall variance is 0.09. Appendix H shows the more details.

The two models trained by Europarl-15 and TED-19 show the same tendency in Figure 2, i.e., the language features for ① of (i) consistently increase in both cases involving the central language, i.e., English, in Figures 2a and 2c, and non-central languages in Figures 2b and 2d. The target language feature of ① emerges as the primary factor that affects representations at the fifth and sixth layers when the cases of (i), (ii), and (iii) are compared. Therefore, we can conclude that the language features of the representations are transferred to the target side within the encoder. Meanwhile, we observe that the scores of (iii) are close to or even exceed those of (ii) at some layers both in Figure 2. This proves that the feature of the source language is not the primary factor for encoding representations because representations are transferred to the subspaces of target languages. Thus, the comparison between (ii) and (iii) supports that language transfer is completed within the encoder.

[3] $(\boldsymbol{x}, l^{①}, \boldsymbol{x})$ indicates the identity of ①, i.e., ①→①, and $(\boldsymbol{y}, l^{①}, \boldsymbol{x})$ indicates a sentence of ② translating to the sentence of ① instructed by the language tag of ①, i.e., ②→①.



(a) Identities. (b) en→x. (c) nl→x.

(d) Identities. (e) en→x. (f) vi→x.

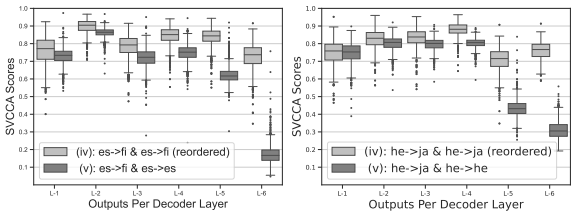
Figure 4: Visualizations for the encoder's output by t-SNE and BiKDE. 4a, 4b and 4c are measured in Europarl-15. 4d, 4e and 4f are measured in TED-19.

On the other hand, identity pairs also allow the measurement of the alignment of different languages in the target language space through t-SNE. Compared with the sentence-level measurement of Pan et al. (2021); Gao et al. (2023), we measure the alignment of representations at the token level. As shown in Figure 3, semantic similarity causes the representations to cluster together. Moreover, as shown in Appendix H, these representations are not clustered before being processed by the encoder, and the case with different target languages has a higher overall variance. Combined with the finding that the encoder transfers the representation of the source language to the target language, the evidence further suggests that there is no general and cross-lingual state for directly sharing semantic information within the encoder, and the alignment shown in Figure 1b occurs in the representational subspace of the target language.

3.3 Entanglements Hindering the Transfer

Although the investigation in Section 3.2 shows that the representations gradually transfer to the target language in a translation pair, the representation spaces of multiple languages may potentially entangle with each other, resulting in the failure of the zero-shot translation (Qu and Watanabe, 2022). To further illustrate the relationship between different languages, we use t-SNE and BiKDE to visualize the representations at the output of the encoder for the several identity pairs in Europarl-15 and TED-19. Figures 4a and 4d show that different identity pairs are uniformly distributed in the representational space. This distribution proves again that the encoder is language-specific because each language has an isolated representational subspace.

Compared with identity pairs that represent the ideal capability of the model in processing languages, the distributions plotted in Figures 4b and 4e reflect the actual capacity for the supervised translation of en→x. Figures 4b and 4e show the distribution of representations in the pairs translating from en, which are similar to that of identity pairs. The difference between identity pairs and supervised language pairs can be attributed to the influence of the source language information, which hinders the full use of the target language information learned by the encoder.



(a) (es, fi) in Europarl. (b) (he, ja) in TED.

Figure 5: Visualizations of layer-wise SVCCA scores for the decoder. (③, ④) shows the involved languages.

Moreover, the language-specific subspaces cannot be clearly separated for zero-shot translations, as shown in Figures 4c and 4f. Specifically, all representations are entangled around the supervised language pair of x→en, which hinders these representations from transferring into the ideal subspaces of the target language. This aligns with Qu and Watanabe (2022) and Stap et al. (2023) that multilingual representations are entangled, which explains the weakness of zero-shot translation compared with supervised translation, suggesting that improving the transferability of representations is attributed to the extent of language transfer within the encoder.

3.4 Language Features in the Decoder

We further investigate the importance of target language features versus semantics in the decoder. Given two sentences $\boldsymbol{x}$ of language ③ and $\boldsymbol{y}$ of language ④, the decoder representation of $(\boldsymbol{x}, l^{④}, \boldsymbol{y})$ is considered as the base measure. We group two cases: (iv) For each sentence in a test set, we identify the pair $(\boldsymbol{x}', l^{④}, \boldsymbol{y}')$ with the lowest SVCCA score in the encoder representation to derive a $\boldsymbol{x}'$ that is distant from $\boldsymbol{x}$. Then, we compare it with the base measure to show the importance of target language features; (v) We compare the base measure and $(\boldsymbol{y}, l^{④}, \boldsymbol{y})$ to show the importance of semantics. The two scenarios shown in Figure 5 present the same trend, which is that (iv) maintains high scores despite their semantics being entirely different. At the top layers of the decoder, the gradually increas-

ing difference between (iv) and (v) confirms that the decoder tends to learn the target language specificity (Sen et al., 2019). However, Figure 5 shows that, for (iv), a wider interquartile range exists at the bottom layers of the decoder, and its scores are close to those of (v), which implies the weakness in distinguishing languages for zero-shot translations.

4 Encouraging Representation Transfer

To validate the findings in Section 3, we propose two methods on the encoder and decoder sides, respectively, to improve transferability. Based on the findings in Sections 3.2 and 3.3, improving the extent of language transfer in the encoder can overcome the hindered representations of zero-shot language pairs. We introduce a learnable embedding referred to as Low-rank Language-specific Embedding (LoLE). It serves as biases to force representations to transfer into the target language with negligible cost. Based on the findings in Section 3.4, the capacity for multilingual features is insufficient at the lower layers of the decoder. We introduce Language-specific Contrastive Learning of Representations (LCLR) as an training extra task to regularize the representations to specify the representational boundary for each language.

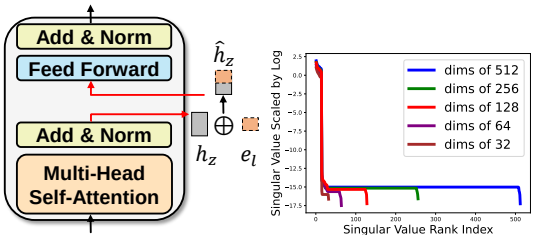
4.1 Low-Rank Embedding for the Encoder

Let $\mathbb{E} = \{\boldsymbol{e}^1, \boldsymbol{e}^2, \dots, \boldsymbol{e}^p\}$, $\boldsymbol{e}^j \in \mathbb{R}^d$, be a set of embeddings that correspond one-to-one with the languages in $\mathbb{L}$. For a translation $(\boldsymbol{x}, l, \boldsymbol{y})$, the embedding in $\mathbb{E}$ corresponding to l is denoted by $\boldsymbol{e}^l$. The hidden representation $\mathbf{H}^z = \{\boldsymbol{h}_1^z, \boldsymbol{h}_2^z, \dots, \boldsymbol{h}_q^z\}$, where $\mathbf{H}^z \in \mathbb{R}^{q \times d}$, is extracted before the feed-forward network (FFN) (Khandelwal et al., 2021; Xu et al., 2023; Deguchi et al., 2023) at the z-th encoder layer. Then, we broadcast $\boldsymbol{e}^l$ to $\mathbf{E}^l$, $\mathbf{E}^l \in \mathbb{R}^{q \times d}$, and we bias $\mathbf{H}^z$ to $\hat{\mathbf{H}}^z$:

$$\hat{\boldsymbol{h}}_i^z = \boldsymbol{h}_i^z + \boldsymbol{e}_i^l, \tag{2}$$

where $\hat{\mathbf{H}}^z$ is the input for the FFN of the z-th encoder layer (Figure 6a). We execute this biasing at the second-top encoder layer to ensure sufficient capacity for fusing representations and language information, while implicitly allowing lower layers to focus on surface-level information.

The simple language categorization by embedding may lead to a risk of dimensional collapse in the latent space (Jing et al., 2022). Thus, we reduce the dimension of $\mathbb{E}$ to d^e to allow biasing in



(a) Process (b) Dimension

Figure 6: Illustrations of LoLE. 6a shows the process in an encoder layer, where $\bigoplus$ represents the operation of Equation(2) at a low rank. 6b is the spectrum with different dimensions of $\mathbb{E}$, which illustrates the singular values of the covariance matrix of $\mathbb{E}$ in sorted order and logarithmic scale.

a low rank, and add it to the head of $\boldsymbol{h}_i^z$ to simultaneously encourage language transfer and minimize the influence on representations (Hu et al., 2021). Figure 6b is a spectrum used to illustrate dimensional collapse using a comparison of different d^e in Europarl-15. The spectrum shows that larger dimensions are primarily composed of noise, whereas a dimension that is too small adversely affects the learning of key features.

4.2 Contrastive Learning for the Decoder

Given a training batch, we extract hidden representations from the output of each decoder layer and apply averaged pooling to obtain a fixed-dimensional representation for each sentence. To avoid dimensional collapse (Tian, 2022; Jing et al., 2022), we also use the head of the representation for contrastive learning, i.e., the vectors in the batch $\mathbb{B} = \{\overline{\boldsymbol{h}}_1, \overline{\boldsymbol{h}}_2, \dots\}$, $\overline{\boldsymbol{h}}_i \in \mathbb{R}^{d^h}$, $d^h < d$.

To prevent a potential invalid training objective in sampling caused by the skewed distribution in a batch, we first define $\mathbb{B}' \subseteq \mathbb{B}$ by omitting instances that do not share their target language with any other instance in $\mathbb{B}$. For a given instance of $\boldsymbol{h}^{\text{anc}} \in \mathbb{B}'$, which is the anchor in contrastive learning, we let $\mathbb{B}^+$ denote the subset of $\mathbb{B}'$, including instances with the same target language as $\boldsymbol{h}^{\text{anc}}$, where $|\mathbb{B}^+| > 1$. Likewise, we define a subset for negative instance $\mathbb{B}^- = \mathbb{B}' \setminus \mathbb{B}^+$. For contrastive learning, we randomly sample the positive instance $\boldsymbol{h}^{\text{pos}}$ from $\mathbb{B}^+$ and sample k negative instances $\boldsymbol{h}^{\text{neg}}$ from $\mathbb{B}^-$. Additionally, if $k > |\mathbb{B}^-|$, we dynamically clip k to $|\mathbb{B}^-|$. Formally, the objective of LCLR is formulated as

$$\begin{aligned} \mathcal{L}_{ctr} &= -\sum_{\boldsymbol{h}^{\text{anc}} \in \mathbb{B}'} \log \frac{e^{s^+}}{e^{s^+} + \sum_{i=1}^{k} e^{s_i^-}}, \\ s^+ &= \text{sim}(\boldsymbol{h}^{\text{anc}}, \boldsymbol{h}^{\text{pos}}), \boldsymbol{h}^{\text{pos}} \in \mathbb{B}^+, \\ s_i^- &= \text{sim}(\boldsymbol{h}^{\text{anc}}, \boldsymbol{h}_i^{\text{neg}}), \boldsymbol{h}_i^{\text{neg}} \in \mathbb{B}^-, \end{aligned} \tag{3}$$

	Europarl-15						TED-19						OPUS-100			
	BLEU			B.S.			BLEU			B.S.			BLEU			B.S.
Method	en→	→en	zero.	en→	→en	zero.	en→	→en	zero.	en→	→en	zero.	en→	→en	zero.	zero.
VANILLA	37.49	43.39	24.65	88.50	95.71	84.27	**24.53**	29.67	11.98	**83.77**	93.54	77.74	23.37	28.30	5.04	69.98
DISPOS	37.15	43.37	25.89	88.39	**95.72**	84.69	24.08	29.43	12.80	83.62	93.49	78.36	22.72	28.24	5.58	70.74
TLP	37.41	43.28	24.96	88.47	95.71	84.40	24.44	29.62	12.74	83.73	93.53	78.24	**23.41**	28.30	4.60	69.40
SEMALI	37.27	43.06	25.25	88.42	95.69	84.43	23.55	28.67	**13.45**†	83.43	93.36	**78.91**†	22.35	28.29	6.42	72.00
LOLE	37.62	43.50	26.09†	**88.51**	**95.72**	84.81	24.39	29.72	13.20	83.74	93.54	78.65	23.15	28.28	7.92†	**73.32**†
LCLR	37.44	43.43	25.71	88.46	**95.72**	84.66	24.46	29.66	12.12	83.76	93.54	77.87	23.34	**28.37**	5.11	70.04
BOTH	**37.67**	**43.51**	**26.20**†	88.50	**95.72**	**84.85**†	24.49	**29.79**	13.31†	83.76	**93.56**	78.76†	23.40	28.27	**7.97**†	73.10†

Table 1: Averaged scores for experiments of training from scratch. BOTH means using LOLE and LCLR together; en→ and →en abbreviates en→x and x→en; zero. means zero-shot language pairs; and B.S. abbreviates BERTScore. We only report zero-shot language pairs of OPUS-100 because BERTScore does not support some pairs in supervised translations, but zero-shot translation pairs of OPUS-100 are involved only with 6 languages, which are supported. The bold number indicates the best result and the numbers with † are significantly better than VANILLA according to the significance test with $p < 0.05$. The off-target ratios are reported in Appendix K.

where $\mathrm{sim}(\cdot)$ calculates the similarity of representations using the cosine similarity. The final training objective is the sum of Equations 1 and 2, i.e.,

$$\mathcal{L} = \mathcal{L}_{ce} + \mathcal{L}_{ctr}. \tag{4}$$

5 Experiments

5.1 Setup

Datasets Our experiments comprise three popular English-centric datasets, i.e., the training and validation sets only involving translation pairs translating to en or from en, including Europarl-15 (Koehn et al., 2005; Dabre and Kurohashi, 2019), TED-19 (Ye et al., 2018) and OPUS-100 (Zhang et al., 2020a; Yang et al., 2021). The details of those datasets can be found in Appendix B.

Evaluation We evaluate the performance of models on the test sets of those three datasets and set the beam size to 4 in inference. We employ SacreBLEU (Papineni et al., 2002; Post, 2018) to evaluate the quality of inferences at the word level and report BERTScore (Zhang et al., 2020b) of inferences at the representation level. We measure the off-target ratio on zero-shot translations as a supplement. We also conduct statistical significance testing (Koehn, 2004). We describe our motivation in selecting evaluation metrics and the evaluation details in Appendix I.

Models When training from scratch, we implement a Transformer model with 6 encoder and decoder layers. Given that those three datasets have different sizes, we set different hyper-parameters in training. Then, three open-source models[4] are utilized in fine-tuning experiments, including M2M-418M, M2M-1.2B (Fan et al., 2020) and mBART50 (Tang et al., 2020). The hyper-parameter settings can be found in Appendix C. Additionally, hyper-parameters are selected based on the ablation studies conducted on the validation sets, which is reported in Appendix F.

Baselines Vanilla Transformer (Vaswani et al., 2017; Johnson et al., 2017) is one of the baselines, denoted by VANILLA. Moreover, three representative methods are reproduced in our experiments of training from scratch, which are denoted by SEMALI (Pan et al., 2021), DISPOS (Liu et al., 2021), and TLP (Yang et al., 2021), respectively. We state our motivation in selecting baselines and the correlation between our proposed methods and those methods in Appendix J.

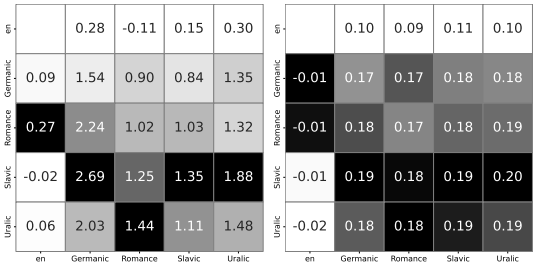
5.2 Results

Table 1 shows the experimental results of training from scratch. In supervised translations of Europarl-15/TED-19/OPUS-100, LOLE shows divergent results of 0.13/-0.04/-0.22 on en→x and 0.11/0.05/-0.02 on x→en. Similarly, LCLR shows diverse results of -0.05/-0.01/-0.03 and 0.04/-0.01/0.07, respectively. Then, BOTH achieves the results of 0.18/-0.02/0.03 on en→x and 0.12/0.12/-0.03 on x→en. In zero-shot translations, BOTH outperforms VANILLA 1.55/1.33/2.93 for BLEU and 0.58/1.02/3.12 for BERTScore. Our models perform best in zero-shot translations of Europarl-15 and OPUS-100, and the improvements in zero-shot translations are always statistically significant. Note that, although SEMALI achieves the best zero-shot translation performance in TED-19, the supervised performance of SEMALI is significantly degraded compared to VANILLA, which is a common

[4]Note that, those models are trained by adding a source language tag at the encoder and a target language tag at the decoder. In fine-tuning, we keep the original strategy.

	BLEU			BERTScore		
Method	en→	→en	zero.*	en→	→en	zero*
M2M-418M	21.88	26.43	14.51	82.52	93.25	79.26
F.T.	26.68	32.95	17.46	84.47	94.30	80.79
LoLE	26.81	33.16	17.52	**84.51**	94.31	80.84
LCLR	26.81	**33.67**†	17.65	84.47	**94.40**	80.88
BOTH	**26.83**	33.63†	**17.68**	84.49	94.38	**80.90**
M2M-1.2B	24.32	28.94	15.95	83.17	93.72	79.75
F.T.	27.71	**34.97**	18.48	84.71	**94.53**	81.14
LoLE	28.29†	34.12	18.67	84.91†	94.48	**81.26**†
LCLR	28.26†	34.54	18.64	84.90†	94.50	81.22
BOTH	**28.37**†	34.59	**18.69**	**84.92**†	94.51	81.23
mBART50	25.28	33.50	6.92	83.93	**94.43**	72.91
F.T.	27.17	33.96	5.58	84.64	94.36	72.96
LoLE	27.19	33.93	7.28†	84.60	94.37	73.86†
LCLR	27.07	34.02	**9.69**†	84.59	94.38	75.31†
BOTH	**27.36**	**34.04**	9.55†	**84.66**	94.36	**75.55**†

Table 2: Averaged scores for experiments of fine-tuning. F.T. means fine-tuning without any trick. * is added to zero. to show it is not a real zero-shot scenario for M2M. The bold number indicates the best result, and the numbers with † are significantly better than F.T.. The off-target ratios are reported in Appendix K.



(a) BLEU (b) SVCCA

Figure 7: Differences between our model and VANILLA. X-axis is the target language family where en is considered solely. Hence, we plot the color ladder by column where the darker the color, the bigger the difference.

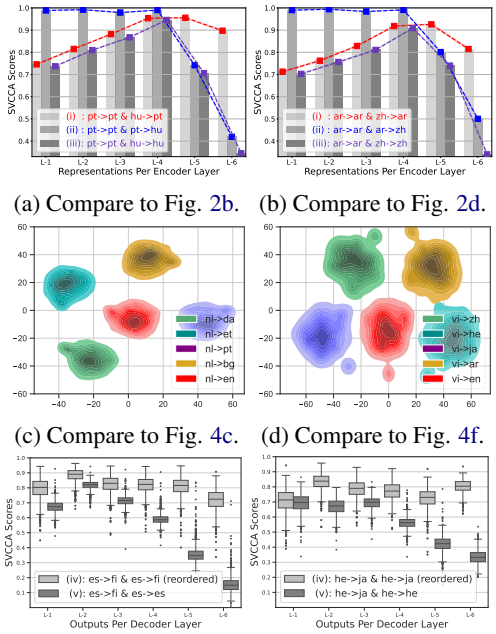
and unresolved problem (Gu et al., 2019; Zhang et al., 2020a; Liu et al., 2021). On the contrary, our methods not only perform competitively with SEMALI in zero-shot translations but also benefit the supervised translation capacity. Moreover, these two proposed methods are orthogonal, which can be proved by assessing LoLE, LCLR and BOTH individually: (1) LoLE achieves gains of 1.53/1.22/2.85 for BLEU and 0.54/0.91/3.34 for BERTScore; (2) LCLR improves 1.06/0.14/0.09 and 0.39/0.13/0.06 scores; (3) The gains of BOTH are always higher than LoLE and LCLR. In addition, we can observe that the improvement of LCLR is limited in TED-19 and OPUS-100, which can be attributed to the diverse languages involving in these two datasets and being easily distinguished by the vanilla decoder. This result also supports that the main challenge of MNMT is the transfer within the encoder. Thus, we can conclude that our methods substantially benefit the zero-shot translation capacity of MNMT models.

Table 2 shows the experimental results of fine-tuning. For M2M-418M, compared with F.T., our methods obtain up to 0.15/0.72/0.22 for BLEU scores and 0.04/0.10/0.11 for BERTScore in en→x, x→en and zero-shot translations, respectively; For M2M-1.2B, the gain is up to 0.66/-0.38/0.21 for BLEU scores and 0.21/-0.02/0.12 for BERTScore; For mBART50, the gain is up to 0.19/0.08/4.11 for BLEU scores and 0.02/0.02/1.69 for BERTScore. Those scores show that the improvement on M2M is marginal compared with training from scratch. This derives M2M is trained by interconnected translation pairs instead of an English-centric dataset, which results in the robust transferability of multilingual representations. However, the degeneration on F.T. of mBART50 shows that fine-tuning drastically influences the zero-shot translation capacity. For instance, the BLUE scores of fr→vi decrease to 11.84 from 20.57 and fr→zh increase to 13.52 from 1.90, but our model obtains 18.47 and 17.19, respectively. Such results and the significant testing indicate again the advantage of our proposed methods in improving multilingual representations for zero-shot translation capacity.

6 Discussion

6.1 Correlation between Representational Disentanglements and Improvements

Table 1 shows the overall results by taking averages across all language pairs, which may overlook pair-specific tendencies. Therefore, we group Europarl-15 by the language families and report the average scores of translating from one language family to another. Figure 7a shows the difference in BLEU scores between our models and VANILLA. As shown in Figure 7b, we also compute the SVCCA scores between the identity of the non-central language and the identity of the central language at the encoder's output and group them in the same manner. Given the similar distribution in Figure 7, we conduct Pearson correlation analysis (Pearson, 1896) of all language pairs instead of language families in Europarl-15, and we compute the coefficients and p-values of Pearson correlation by target languages to maintain fairness. We observe two key points: 1) The coefficient and p-value of en are -0.087 and 0.76, respectively. This result suggests that there is no statistical correlation, which is predictable because x→en is not affected by representational entanglements. 2) The coefficient and p-value of non-central languages are in the ranges



(a) Compare to Fig. 2b. (b) Compare to Fig. 2d.

(c) Compare to Fig. 4c. (d) Compare to Fig. 4f.

(e) Compare to Fig. 5a. (f) Compare to Fig. 5b

Figure 8: Visualizations for the encoder that incorporates LoLE and LCLR, showing improvements compared with VANILLA. Additionally, the model plotted in 8e only incorporates LCLR.

of 0.585 to 0.855 and 4e-5 to 0.021, respectively. In more detail, the mean values are 0.770 and 0.002 and the variances are 0.04 and 3e-5, respectively. This analysis proves that the degree of representational disentanglement positively correlates with the improvement of zero-shot translations.

6.2 Analysis of Improved Representation

We measure representation transfer in the model incorporating our proposed methods to verify our findings further. As shown in Figures 8a and 8b, both scenarios exhibit improvements on (i). Meanwhile, Figures 8c and 8d indicate that the entanglement of representations among languages at the encoder is resolved. The evidence suggests that LoLE effectively enhances representation transfer in the encoder. Additionally, (ii) and (iii) in Figures 8a and 8b also achieve higher scores at lower layers of the encoder, which suggests that LoLE indeed makes lower layers of the encoder focus on surface-level information. By contrast, as shown in Figures 8e and 8f, the more stable trend of (iv) in both scenarios suggests that LCLR can improve the capacity of lower layers of the decoder to distinguish languages to improve zero-shot translations. In addition, Appendix G provides the representational analysis for fine-tuning models, which proves that target language features are consistently beneficial in the encoder.

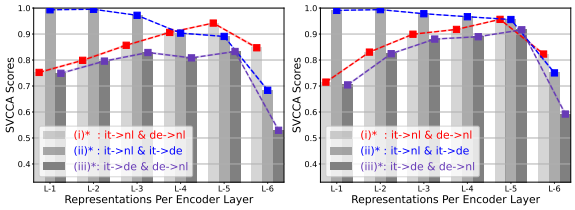
7 Related Works

Prior studies on analyzing multilingual representation in Section 2.2 led to several effective methods in MNMT. Some works focused on updating and constraining the encoder to improve multilingual representations, and the findings in discrepancy mentioned in Section 2.2 led to two distinct approaches. First, Pan et al. (2021); Gu and Feng (2022); Gao et al. (2023); Bu et al. (2024) suggested regularizing the encoder for aligning semantic information across different source languages by introducing additional training objectives. Similarly, Pham et al. (2019); Zhu et al. (2020) explicitly modified the output form of the encoder to transfer the representation of the source sentence toward a language-agnostic state. Second, Gu et al. (2019); Liu et al. (2021); Sun et al. (2024) introduced specialized modeling constraints to improve the encoder to transfer source sentence representations to the target language without adding extra parameters, and Zhang et al. (2021); Pires et al. (2023) enhanced the representation of target language information by simply adding language-specific modules. Additionally, Yang et al. (2021); Qu and Watanabe (2022); Bu et al. (2024) focused on improving the target language representation on the decoder side or adding modules specified to the target language to the decoder. Given that the above works can all be encompassed within our analyses, we argue that this work offers insights for future improvements in MNMT. Specifically, enhancing the encoder to transfer source language representations into the target language subspace and align semantic information within those subspaces is the key to improving MNMT.

8 Conclusion

We systematically investigated the representational issue of zero-shot translation deficiency in multilingual neural machine translation models. Our analyses show that the encoder transfers translation representations from the source language to the target language, and aligns semantics across different source languages at the target language subspace. We applied engineering practices to verify our findings by proposing two orthogonal methods, which substantially improve the zero-shot translation capacity. Thus, our findings are significant for guiding the improvement of the transferability of multilingual representations.

9 Limitations



(a) Bidirection (b) Unidirection

Figure 9: Illustration of the comparison between the bidirectional and the unidirectional scenarios. 9a has the same model settings with Figure 2, but analyzes the same pairs with 9b.

This work has two limitations. First, the identity pair is a proxy of language representations based on bi-directional training, i.e., each non-central language appears in the encoder and decoder together. Therefore, we designed an additional study to investigate the impact by retraining a model by eliminating `nl`→`en` and `en`→`it` so that `it` and `nl` appear only in the encoder and decoder, respectively, based on the analysis in Section 3.2. Then, we conducted a comparison by taking de as the middle language to perform the role of identity pairs in analysis. As shown in Figure 9, the target language features keep the same trend as shown in Section 3.2 to support our conclusion again, but the influence of source language features increases relatively. The second limitation is our investigation is based on adding a language tag specified to the target language at the beginning of the source sentence for the encoder. Although this is the de facto MNMT training strategy (Johnson et al., 2017; Aharoni et al., 2019; Arivazhagan et al., 2019a; Gu et al., 2019; Pham et al., 2019; Wu et al., 2021; Yang et al., 2021; Pan et al., 2021; Qu and Watanabe, 2022; Chen et al., 2023; Gu and Feng, 2022; Gao et al., 2023), the current open-source models (Fan et al., 2020; Tang et al., 2020; Team et al., 2022) are based on another strategy, i.e., adding a source language tag at the encoder side and adding a target language tag at the decoder side. Although, in Section 5, we have shown our proposed methods also benefit models with this strategy, this effectiveness is proved by empirical experiments. Thus, our future work is to investigate the representation transfer of this strategy to guide further improvements.

10 Ethical Considerations

All datasets used in this work are public data, which are proven harmless. Moreover, this work is foundational research and is not tied to particular applications. Thus, there is no ethical risk existed in this work.

References

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. *Preprint*, arXiv:1903.00089.

Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Roee Aharoni, Melvin Johnson, and Wolfgang Macherey. 2019a. The missing ingredient in zero-shot neural machine translation. *Preprint*, arXiv:1903.07091.

Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Dmitry Lepikhin, Melvin Johnson, Maxim Krikun, Mia Xu Chen, Yuan Cao, George Foster, Colin Cherry, Wolfgang Macherey, Zhifeng Chen, and Yonghui Wu. 2019b. Massively multilingual neural machine translation in the wild: Findings and challenges. *Preprint*, arXiv:1907.05019.

Mengyu Bu, Shuhao Gu, and Yang Feng. 2024. Improving multilingual neural machine translation by utilizing semantic and linguistic features. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 10410–10423, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Liang Chen, Shuming Ma, Dongdong Zhang, Furu Wei, and Baobao Chang. 2023. On the off-target problem of zero-shot multilingual neural machine translation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9542–9558, Toronto, Canada. Association for Computational Linguistics.

Raj Dabre and Sadao Kurohashi. 2019. Mmcr4nlp: Multilingual multiway corpora repository for natural language processing. *Preprint*, arXiv:1710.01025.

Hiroyuki Deguchi, Taro Watanabe, Yusuke Matsui, Masao Utiyama, Hideki Tanaka, and Eiichiro Sumita. 2023. Subset retrieval nearest neighbor machine translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 174–189, Toronto, Canada. 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. *Preprint*, arXiv:2010.11125.

Orhan Firat, Baskaran Sankaran, Yaser Al-onaizan, Fatos T. Yarman Vural, and Kyunghyun Cho. 2016. Zero-resource translation with multi-lingual neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language*

Processing, pages 268–277, Austin, Texas. Association for Computational Linguistics.

Pengzhi Gao, Liwen Zhang, Zhongjun He, Hua Wu, and Haifeng Wang. 2023. Improving zero-shot multilingual neural machine translation by leveraging cross-lingual consistency regularization. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12103–12119, Toronto, Canada. 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.

Shuhao Gu and Yang Feng. 2022. Improving zero-shot multilingual translation with universal representations and cross-mapping. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6492–6504, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

David R. Hardoon, Sandor Szedmak, and John Shawe-Taylor. 2004. Canonical correlation analysis: An overview with application to learning methods. *Neural Computation*, 16(12):2639–2664.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. 2022. Understanding dimensional collapse in contrastive self-supervised learning. *Preprint*, arXiv:2110.09348.

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.

Urvashi Khandelwal, Angela Fan, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2021. Nearest neighbor machine translation. In *International Conference on Learning Representations*.

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A method for stochastic optimization. *Preprint*, arXiv:1412.6980.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395, Barcelona, Spain. Association for Computational Linguistics.

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.

Sneha Kudugunta, Ankur Bapna, Isaac Caswell, and Orhan Firat. 2019. Investigating multilingual NMT representations at scale. 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 1565–1575, Hong Kong, China. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *Preprint*, arXiv:1910.13461.

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.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Preprint*, arXiv:2001.08210.

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.

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.

Karl Pearson. 1896. Mathematical contributions to the theory of evolution. iii. regression, heredity, and panmixia. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 187:253–318.

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.

Telmo Pires, Robin Schmidt, Yi-Hsiu Liao, and Stephan Peitz. 2023. Learning language-specific layers for multilingual machine translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14767–14783, Toronto, Canada. 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.

Zhi Qu and Taro Watanabe. 2022. Adapting to non-centered languages for zero-shot multilingual translation. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 5251–5265, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. 2017. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6078–6087, Red Hook, NY, USA. Curran Associates Inc.

Naomi Saphra and Adam Lopez. 2019. Understanding learning dynamics of language models with SVCCA. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3257–3267, Minneapolis, Minnesota. Association for Computational Linguistics.

Sukanta Sen, Kamal Kumar Gupta, Asif Ekbal, and Pushpak Bhattacharyya. 2019. Multilingual unsupervised NMT using shared encoder and language-specific decoders. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3083–3089, Florence, Italy. Association for Computational Linguistics.

David Stap, Vlad Niculae, and Christof Monz. 2023. Viewing knowledge transfer in multilingual machine translation through a representational lens. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14973–14987, Singapore. Association for Computational Linguistics.

Zengkui Sun, Yijin Liu, Fandong Meng, Jinan Xu, Yufeng Chen, and Jie Zhou. 2024. LCS: A language converter strategy for zero-shot neural machine translation. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 9201–9214, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Shaomu Tan and Christof Monz. 2023. Towards a better understanding of variations in zero-shot neural machine translation performance. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13553–13568, Singapore. Association for Computational Linguistics.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2020. Multilingual translation with extensible multilingual pretraining and finetuning. *Preprint*, arXiv:2008.00401.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation. *Preprint*, arXiv:2207.04672.

Yuandong Tian. 2022. Understanding deep contrastive learning via coordinate-wise optimization. In *Advances in Neural Information Processing Systems*.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605.

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.

M. P. Wand and M. C. Jones. 1993. Comparison of smoothing parameterizations in bivariate kernel density estimation. *Journal of the American Statistical Association*, 88(422):520–528.

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.

Frank F. Xu, Uri Alon, and Graham Neubig. 2023. Why do nearest neighbor language models work? *Preprint*, arXiv:2301.02828.

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, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Qi Ye, Sachan Devendra, Felix Matthieu, Padmanabhan Sarguna, and Neubig Graham. 2018. When and why are pre-trained word embeddings useful for neural machine translation. In *HLT-NAACL*.

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. 2020a. 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.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020b. Bertscore: Evaluating text generation with bert. *Preprint*, arXiv:1904.09675.

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, Online. Association for Computational Linguistics.

A Sentence-level SVCCA Score

We use SVCCA (Raghu et al., 2017) to measure representation similarity in MNMT (Kudugunta et al., 2019). We follow the approach of Liu et al. (2021) so that similarity is measured at the sentence level to ensure that each score is computed on equivalent features without the influence of other sentences in the set.

Based on the definition of Section 2.1, we denote hidden representations of a sentence by $\mathbf{H} = \{\boldsymbol{h}_1, \boldsymbol{h}_2 ... \boldsymbol{h}_q\}$, where $\mathbf{H} \in \mathbb{R}^{q \times d}$, q equals to the length n or m from either the encoder or decoder, and d is the model dimension. Additionally, the practical length is $n+1$ when $\mathbf{H}$ is fed into the encoder because the encoder receives the input concatenated by l and $\boldsymbol{x}$[5]. Then, we derive the sentence-level representation $\overline{\boldsymbol{h}}$ using average pooling $\overline{\boldsymbol{h}} = \frac{\sum_{i=1}^{q} \boldsymbol{h}_i}{q}$, which mainly represents the language features and semantics of the source sentence rather than syntactic information because positional information is reduced.

Given $\mathbf{H}^a$ and $\mathbf{H}^b$ derived from two sentences, SVCCA first performs singular value decomposition on their averaged representations to obtain subspace representations $\overline{\boldsymbol{h}}^a \in \mathbb{R}^{d^a}$ and $\overline{\boldsymbol{h}}^b \in \mathbb{R}^{d^b}$, where noise is reduced (Saphra and Lopez, 2019). Then we perform canonical correlation analysis (Hardoon et al., 2004) to determine $\mathbf{W}^a \in \mathbb{R}^{d' \times d^a}$ and $\mathbf{W}^b \in \mathbb{R}^{d' \times d^b}$. Formally, we compute correlation ρ between $\overline{\boldsymbol{h}}^a$ and $\overline{\boldsymbol{h}}^b$ as

$$\rho = \frac{\langle \mathbf{W}^a \overline{\boldsymbol{h}}^a, \mathbf{W}^b \overline{\boldsymbol{h}}^b \rangle}{\|\mathbf{W}^a \overline{\boldsymbol{h}}^a\| \|\mathbf{W}^b \overline{\boldsymbol{h}}^b\|}, \tag{5}$$

where $\langle \cdot, \cdot \rangle$ indicates the inner product. We use ρ to represent the similarity of two sentences. Finally, we compute the set-level score by taking the average scores of all sentences over the set.

B Detailed Information of Datasets

This work involves three datasets, i.e., Europarl-15, TED-19, and OPUS-100, where Europarl-15 and TED-19 are used in preliminary experiments. The training sets of those three datasets have different sizes, but the validation and test sets of a pair generally contain 2,000 translation instances. In preliminary experiments, we measure SVCCA scores in the test sets because those instances are unseen in the training.

Europarl-15 is collected from MMCR4NLP, which has high-quality translation instances and each instance in a language is one-to-one corresponding to other languages, i.e., all language-specific sets have parallel semantics (Koehn et al., 2005; Dabre and Kurohashi, 2019), including 15 European languages from 4 language families. Specifically, Germanic includes `en`, `de`, `nl`, `da`, Romance includes `es`, `pt`, `it`, `ro`, Slavic includes `sl`, `bg`, `pl`, `cs`, and Uralic includes `fi`, `et`, `hu`. The training and validation sets cover 28 supervised translation pairs where English is the central language used to bridge the non-central languages. The test set consists of all language pairs, including 182 zero-shot translation pairs in addition to supervised translation pairs. Finally, each pair in the training set comprises 189,310 instances.

In contrast to Europarl-15, which is the semantically aligned dataset, TED-19 consists of 19 lan-

[5] l plays the role of translation instruction instead of a token belonging to the target language with semantics, thus, this concatenation would not influence the measurement by mixing target language information into the sentence representation within the encoder.

guages, including en, ar, he, ru, ko, it, ja, zh, es, nl, vi, tr, fr, pl, ro, fa, hr, cs, de, which belong to various language families without parallel semantics, from TED Talks (Ye et al., 2018). Each translation pair contains 103,093 to 214,111 instances in training, and the training set comprises 6,551,456 instances in total. Because of the unparallel semantics of TED-19, we align ar, he, zh, hr, vi, ja to obtain 967 translation instances for measuring SVCCA scores. In addition, the reason why the number of languages is 19 is that, first, TED Talks have 20 high-resource languages, which are supported in M2M (Fan et al., 2020) and mBART50 (Tang et al., 2020). However, the tokenization of th is problematic, resulting in deprecating th.

OPUS-100 consists of 95 languages, 188 pairs, and 109.2 million instances in total (Zhang et al., 2020a; Yang et al., 2021), where 90 pairs comprise 1 million instances and 56 pairs have more than 0.1 million instances. Different from Yang et al. (2021), we do not include the zero-shot translation pairs in the validation set to avoid biases when assessing the transferability of multilingual representations.

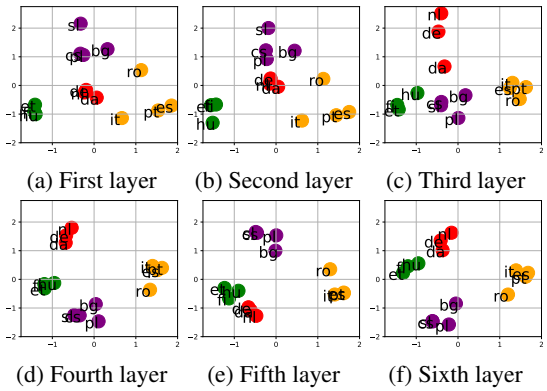
C Detailed Settings of Models

We implement the Transformer (Vaswani et al., 2017) as the backbone model via Fairseq (Ott et al., 2019). For the configuration of models trained on Europarl-15 and TED-19, we follow Kudugunta et al. (2019) to set 6 encoder and decoder layers. Based on the ablation study conducted in the validation set in Europarl-15 and TED-19 shown in Appendix F, we apply LoLE in the fifth encoder layer, set d^e to 128, and set d^h and k of LCLR to 64 and 30, respectively. When we solely apply LCLR, we set the position to the bottom decoder layer based on the findings in Section 3.4. When we integrate both LoLE and LCLR into a model, we relocate LCLR to the second-bottom decoder layer because of the improved language features of the encoder representations. In addition, we investigate models in Appendix E with 8 and 10 encoder layers. We adopt a shared vocabulary trained by SentencePiece (Kudo and Richardson, 2018) with 50,000 tokens for both the encoder and decoder. The model consists of 4 attention heads, embedding size of 512, inner size of 1024, dropout rate of 0.2, maximum learning rate of 0.0005 with the inverse square root schedule and 4,000 warmup steps, and label smoothing rate of 0.1. We set the batch size to 8,000 tokens per GPU, apply Adam (Kingma and Ba, 2017) as the optimizer, and set temperature sampling with $T = 5$ (Arivazhagan et al., 2019b). We train the model with 60 epochs for Europarl-15 and 30 epochs for TED-15, and finally average the top 5 checkpoints using the loss on the validation set. Compared with the basic configuration, the models trained on OPUS-100 have 8 attention heads, embedding size of 512, inner size of 2048, dropout rate of 0.1, and shared vocabulary size of 64,000. We enlarge d^e to 256 and d^h to 128 for models trained on OPUS-100 and three pre-trained models because they involve more languages. We train the model of OPUS-100 for 400,000 update steps with a batch size of 8,000 tokens per GPU for OPUS-100 and directly use the best checkpoint selected using the loss on the validation set. Furthermore, models with Europarl-15 and TED-19 are trained on 8 NVIDIA V100 GPUs, and models with OPUS-100 are trained on 4 NVIDIA A6000 GPUs by setting *–update-freq* to 2 in Fairseq to simulate 8 GPUs.

Three open-source models are utilized in fine-tuning experiments. The first is M2M-418M (Fan et al., 2020), trained on standard multilingual translation tasks and supporting translation across 100 languages. It is based on Transformer architecture, configured with 12 encoder and decoder layers, embedding size of 1024, inner size of 4096, and vocabulary size of 128,112, which results in a total of 418 million parameters. The second model, M2M-1.2B (Fan et al., 2020), enlarges the number of layers to 24 and the inner size to 8192 on M2M-418M, and culminates in 1.2 billion parameters. The last model is mBART50 (Tang et al., 2020), trained on monolingual corpora across 50 languages following Lewis et al. (2019); Liu et al. (2020) and preliminarily fine-tuned for MNMT. It shares the same parameter setup as M2M-418M with a vocabulary size of 250,053, which consists of 611 million parameters. We conduct experiments on TED-19 because all covered languages are supported by these models.

D Detailed Introductions of Figure 1

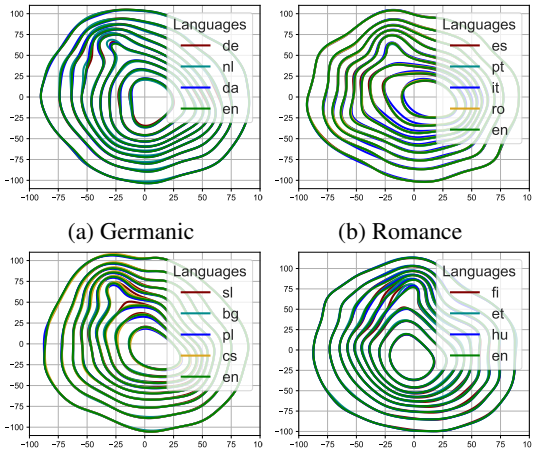
In fact, Figure 1a corresponds to the last subfigure of Figure 10 to show the linguistic affinity between translations from English to other languages, denoted by en→x. Specifically, Figure 10 shows the layer-wise states of the encoder, and Figure 1a (Figure 10f) demonstrates



(a) First layer (b) Second layer (c) Third layer

(d) Fourth layer (e) Fifth layer (f) Sixth layer

Figure 10: Affinities for en→x at each encoder layer. Language families of Europarl-15 are distinguished by colors: Germanic by red, Romance by yellow, Slavic by purple, and Uralic by green.



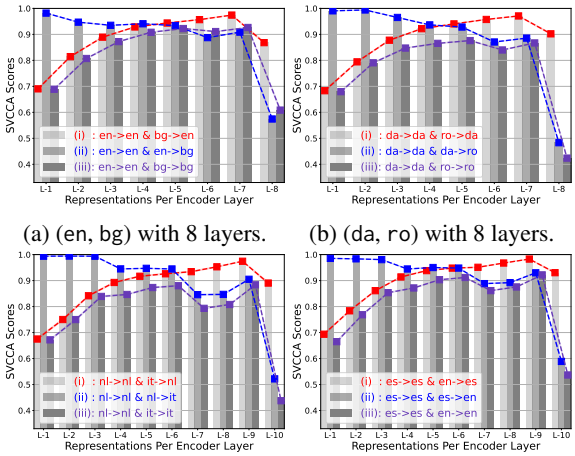
(a) Germanic (b) Romance

(c) Slavic (d) Uralic

Figure 11: Visualizations by t-SNE and BiKDE of aligning representations between en→en and x→en of Europarl-15 at the output of the encoder.

the state at the output of the encoder. We employ *sklearn.manifold.SpectralEmbedding*, referring to https://scikit-learn.org, to visualize the similarities computed by SVCCA (Appendix A) for every layer in the encoder. Then, we can find that representations at all encoder layers have certain clusters influenced by the families of the target languages, and the clusters become more distinct as the depth of the encoder layers increases. This suggests that the transfer of representations to the target language begins as early as the first layer of the encoder, with gradual strengthening through further layers. Meanwhile, this finding, i.e., even the initial encoder layers capture target language features, complements prior works (Kudugunta et al., 2019; Pires et al., 2023).

On the other hand, we follow Pan et al. (2021)



(a) (en, bg) with 8 layers. (b) (da, ro) with 8 layers.

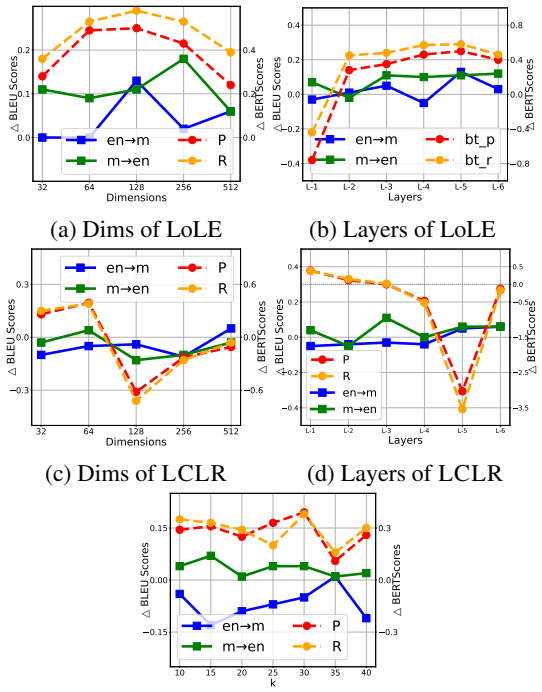
(c) (nl, it) with 10 layers. (d) (es, en) with 10 layers.

Figure 12: Visualizations of 8 and 10 encoder layers on Europarl-15 languages, which are different from Figure 2 as a comparison to prove the generalization.

and Gao et al. (2023) to measure the alignment of encoder representations between the identity of en and source languages from different families to English using t-distributed stochastic neighbor embedding (t-SNE) (van der Maaten and Hinton, 2008) and bivariate kernel density estimation (BiKDE) (Wand and Jones, 1993). As shown in Figure 11, representations from the four language families are all highly aligned with the identity pair of en→en, where the common feature of those translations is the parallel semantics. Thus, this proves that the encoder semantically aligns different translations. However, the deep discussion should be referred to Section 3.2.

E Investigation on Models with 8 and 10 Encoder Layers

Figure 12 shows the scenarios of 8 and 10 encoder layers to confirm our conclusion, where all the cases for (i) consistently reflect the same tendency. Together with the increased number of layers, the increased final scores of (i) indicate the stronger target language features of ① presented for representations. Meanwhile, we also observe that the scores of (iii) are close to or even exceed those of (ii) at some layers both in Figures 2 and 12. This proves that the feature of the source language is not the primary factor for representations in the encoder because representations are transferred to the subspaces of target languages. Thus, this further investigation supports our conclusion that language transferred within the encoder is general.



(a) Dims of LoLE (b) Layers of LoLE

(c) Dims of LCLR (d) Layers of LCLR

(e) k of LCLR

Figure 13: Illustrations for the ablation study. Δ means the difference between the scores of our methods and the scores of VANILLA. 13a and 13b present variations of LOLE in dimensions and layers, respectively; and 13c, 13d, and 13e present variations of LCLR in dimensions, layers, and k, respectively.

F Selecting Hyper-Parameters

We conduct ablation studies on the validation set of Europarl-15 to select hyper-parameters for LOLE and LCLR, which are used in Section 5.1. Figure 13a shows that LOLE performs optimally with the dimension of 128, which corroborates our hypothesis in Section 4.1. Figure 13b indicates that LOLE performs the best at the fifth layer and degrades significantly at lower layers, which aligns with our assertion in Section 3.2 that lower layers of the encoder are more correlated with the source language, and enhances the language transfer benefits of transferability (Section 4.1). Figure 13c is consistent with the theory of contrastive learning in which full dimensions lead to collapse (Jing et al., 2022). Figure 13d demonstrates that, as the position constructed by LCLR increases, the scores decrease, which lends support to our analysis in Sections 3.4 that the instability of decoder representations primarily manifests at lower layers, which also explains the weakness of TLP because improving the capacity of distinguishing languages is redundant for the decoder's top layer. We also conduct an ablation study for hyperparameter k for LCLR with a dimension of 64 at the bottom decoder layer. The results are shown in Figure 13e, with an empirically optimal $k = 30$.

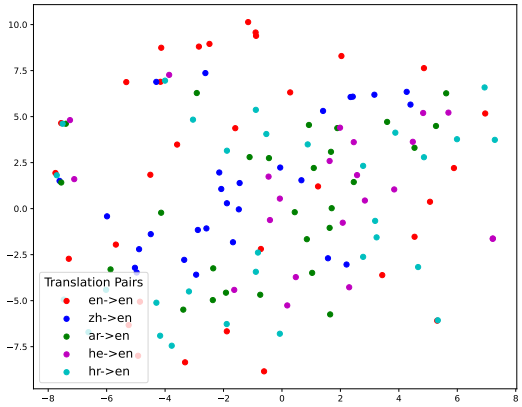
	Pairs	Model	Method	(i)	(ii)	(iii)
Encoder Side	① of zh ② of ar	M2M	F.T.	79.66	100.0	79.66
			LOLE	79.52	98.75	77.76
		mBART	F.T.	63.97	100.0	63.97
			LOLE	63.52	97.90	61.66
	① of he ② of vi	M2M	F.T.	81.50	100.0	81.50
			LOLE	80.54	98.27	79.88
		mBART	F.T.	69.17	100.0	69.17
			LOLE	70.46	97.61	67.04
Decoder Side	① of ja ② of he	M2M	F.T.	99.80	92.01	92.65
			LOLE	99.73	89.81	90.66
		mBART	F.T.	98.53	90.76	89.62
			LOLE	98.64	90.07	88.49

Table 3: SVCCA scores. Each score times 100 for a clear illustration. (i) compares the identity of ① and ①→②, (ii) compares the identity of ① and ②→①, and (iii) compares identities of ① and ②. Encoder Side means computing the output of the encoder, and Decoder Side means computing the output of the 1st layer of the decoder.

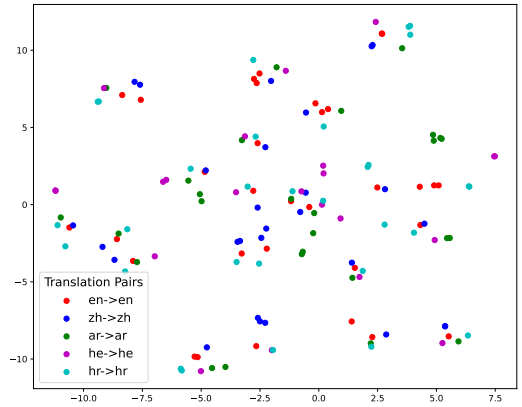
G Analysis of Improved Representation for Fine-tuning Pre-trained Models

Section 6.2 is the representational analysis for models, which are trained from scratch with proposed LOLE and LCLR. We also show the representational analysis for fine-tuning pre-trained models.

Given the positive correlation shown in Section 6.1, we compute SVCCA scores in the same way as done in Section 3.2 and show the results in Table 3. Unlike Section 3.2, we equally consider the encoder and decoder because the encoder is only related to the source language and does not transfer representations to the target language in the training strategy of M2M (Fan et al., 2020) and mBART50 (Tang et al., 2020). Additionally, the different training strategy is the primary reason that F.T. shows the same scores in (i) and (iii) and keeps 100.0 in (ii). Alternatively, although the scores of (i), which reflect target language features, decrease in our methods, the scores of (ii) and (iii) also decrease. As a result, the differences between the scores of (i), (ii), and (iii) increase, that is, the relative importance of target language features increases. This result proves our statements in Sections 3.3 and 6.1 again that target language features are consistently beneficial in the encoder. On the other hand, the decoder side shows the same tendency as the encoder side. This fits our motivation in Section



(a) Semantic alignments on embeddings



(b) Semantic alignments on identities

Figure 14: Illustration of the token-level alignment corresponding to Figure 1b. Representations shown in 14a are collected at the embedding layer, whose overall variance is 1.45. Representations shown in 14b are collected from identities, whose overall variance is 0.13.

4.2 to further improve the discriminating ability of lower layers of the decoder, although the training strategy of M2M and mBART50 has already provided a high capacity in discrimination for the decoder.

H Token-level Alignments in Other Cases

First of all, the English sentence for semantic analysis in Figures 3 and 14 is: By the end of this year, there will be nearly a billion people on this planet that actively use social networking sites. Compared with the discussion in Section 3.2, token-level representations are not aligned at the embedding layer, and are relatively aligned in the case of using the identity pairs, where the degree of divergence is substantially higher than the case of Figure 3.

I Evaluation Metrics Selection

In this work, we select two main automatic evaluation metrics and a secondary statistic measurement. The first one is SacreBLEU (Post, 2018) which is an implementation of BLEU (Papineni et al., 2002). This is the most popular and common metric used in evaluating the alignment between inferences and references at the word level. In order to counter the insufficiency of SacreBLEU, we also select BERTScore (Zhang et al., 2020b), which is a representational metric to evaluate the semantic similarity between inferences and references. Furthermore, to show whether the improvements brought by proposed methods are significant, we also conduct the statistical significance testing (Koehn, 2004) using paired bootstrap resampling with 1,000 iterations and 0.5 resampling ratios, consequently, the case of $p < 0.05$ means that the difference is significant.

Additionally, we follow prior works (Yang et al., 2021; Chen et al., 2023) to report the off-target ratio, which is measured by *fasttext-langdetect*[6]. The off-target translation refers to a sentence translated to an incorrect target language rather than the target language we expected. However, the off-target ratio is not reliable, because the popular tools used in measuring off-target ratios are based on word level and lack support in low-resource languages. Furthermore, the score of SacreBLEU can directly show the problem of off-target, because the evaluation process of SacreBLEU tends to give a great penalty on an inference, which has a different writing script from the expected target language. Therefore, we only report it as a secondary metric in Appendix K.

J Baselines Selection

First of all, our investigation in Section 3.3 reveals that the zero-shot translation deficiency of the vanilla Transformer model is caused by the poor transferability in multilingual representations. In proposing methods, our motivation is to improve the transferability by regularizing representations without additional parameters to validate our conclusions. As a result, LoLE is proposed to bias source representations to the target language on the encoder; LCLR is proposed to encourage the decoder to focus on distinguishing different languages. Thus, besides VANILLA, we select three methods, which are mostly related to our methods:

- SEMALI is proposed by Pan et al. (2021). They think the encoder output is language-

[6] https://pypi.org/project/fasttext-langdetect

	Europarl-15		TED-19		OPUS-100	
Method	zero.(↑)	off.(↓)	zero.(↑)	off.(↓)	zero.(↑)	off.(↓)
VANILLA	24.65	1.34	11.98	4.08	5.04	70.41
DISPOS	25.89	0.84	12.80	3.82	5.58	61.65
TLP	24.96	1.22	12.74	3.71	4.60	83.29
SEMALI	25.25	0.99	13.45	3.62	6.42	58.25
LOLE	26.09	0.71	13.20	3.69	7.92	50.05
LCLR	25.71	0.79	12.12	3.86	5.11	68.53
BOTH	26.20	0.74	13.31	3.69	7.97	55.06

Table 4: Off-target ratio corresponding to experimental results in Table 1. zero. indicates the BLEU scores of zero-shot translations. off. indicates the off-target ratio counted by all zero-shot translation pairs.

Model	Metric	PRE.	F.T.	LOLE	LCLR	BOTH
M2M-418M	zero.(↑)	14.51	17.46	17.52	17.65	17.68
	off.(↓)	3.66	3.34	3.32	3.24	3.33
M2M-418M	zero.(↑)	15.95	18.48	18.67	18.64	18.69
	off.(↓)	3.50	3.15	3.16	3.16	3.14
mBART50	zero.(↑)	6.92	5.58	7.28	9.69	9.55
	off.(↓)	43.56	65.26	40.76	38.24	35.28

Table 5: Off-target ratio corresponding to experimental results in Table 2. Abbreviations follow Table 2 and PRE. refers to the model without any fine-tuning. In addition, compared to Table 4, we switched the horizontal and vertical axes, because there is only one dataset, TED-19, used in fine-tuning experiments.

agnostic, so they align the semantic information across different languages at the output of the encoder. However, our analysis shows that this viewpoint is inaccurate because the semantic information is aligned by the subspace of the target language instead of the real language-agnostic. When there are not any additional parameters introduced, SEMALI still is the de-facto SOTA based on regularizing representations in MNMT.

- DISPOS is proposed by Liu et al. (2021). They have the same objective as LoLE, however, they suggest reducing the constraint on the encoder (Gu et al., 2019) by removing the residual connection, which is a different style that corresponds to the idea of biasing we used in LoLE.

- TLP is proposed by Yang et al. (2021). This method aims to add a loss to predict the language id at the top layer of the decoder, which is contrary to LCLR and our analysis in Section 3.4 where we argue that the bottom layers of the decoder are more sensitive to the language features.

K Off-Target Ratio of Results

Tables 4 and 5 show the measurement of the off-target ratio, which are the supplement of Tables 1 and 2. We can observe that the off-target ratio is always inversely proportional to BLEU scores, aligning with our discussion in Appendix I. Additionally, there are two points worth noting: (1) In Table 4, the off-target ratio in OPUS-100 is generally higher. This is not an outlier because resulting in a strong zero-shot translation capability in OPUS-100 is particularly challenging due to the large number of languages involved and the limited corpus for individual languages (Zhang et al., 2020a; Yang et al., 2021). (2) In Table 5, the off-target ratio counted from mBART50 is higher than other cases. This abnormal value has been discussed in Section 5.2, that is, the zero-shot ability of mBART50 is weaker than M2M models, and then, the fine-tuning dramatically changes the behavior of the model.