Improving Zero-Shot Multilingual Translation with Universal Representations and Cross-Mappings

Shuhao Gu[1,2], **Yang Feng**[1,2*]
[1] Key Laboratory of Intelligent Information Processing,
Institute of Computing Technology, Chinese Academy of Sciences (ICT/CAS)
[2] University of Chinese Academy of Sciences
{gushuhao19b,fengyang}@ict.ac.cn

Abstract

The many-to-many multilingual neural machine translation can translate between language pairs unseen during training, i.e., zero-shot translation. Improving zero-shot translation requires the model to learn universal representations and cross-mapping relationships to transfer the knowledge learned on the supervised directions to the zero-shot directions. In this work, we propose the state mover's distance based on the optimal theory to model the difference of the representations output by the encoder. Then, we bridge the gap between the semantic-equivalent representations of different languages at the token level by minimizing the proposed distance to learn universal representations. Besides, we propose an agreement-based training scheme, which can help the model make consistent predictions based on the semantic-equivalent sentences to learn universal cross-mapping relationships for all translation directions. The experimental results on diverse multilingual datasets show that our method can improve consistently compared with the baseline system and other contrast methods. The analysis proves that our method can better align the semantic space and improve the prediction consistency.

1 Introduction

The many-to-many multilingual neural machine translation (NMT) (Ha et al., 2016; Firat et al., 2016; Johnson et al., 2017; Gu et al., 2018; Fan et al., 2020; Zhang et al., 2020a) model can support multiple translation directions in a single model. The shared encoder encodes the input sentence to the semantic space, and then the shared decoder decodes from the space to generate the translation of the target language. This paradigm allows the model to translate between language pairs unseen during training, i.e., zero-shot translation. Zero-shot translation can improve the inference efficiency and make the model require less bilingual training data. Performing zero-shot translation requires universal representations to encode the language-agnostic features and cross-mapping relationships that can map the semantic-equivalent sentences of different languages to the particular space of the target language. In this way, the model can transfer the knowledge learned in the supervised translation directions to the zero-shot translation directions. However, the existing model structure and training scheme cannot ensure the universal representations and cross-mappings because of lacking explicit constraints. Specifically, the encoder may map different languages to different semantic subspaces, and the decoder may learn different mapping relationships for different source languages, especially when the model possesses high capacity.

Many researchers have made their attempts to solve this problem. Pham et al. (2019) propose to compress the output of the encoder into a consistent number of states to only encode the language-independent features. Arivazhagan et al. (2019) add a regularizing loss to maximize the similarities between the sentence representations of the source and target sentences. Pan et al. (2021) propose contrastive learning schemes to minimize the sentence representation gap of similar sentences and maximize that of irrelevant sentences. All the above work tries to minimize the representation discrepancies of different languages at the sentence level, bringing two problems for NMT. Firstly, these work usually get the sentence-level representation of the encoder output by max-pooling or averaging, which may potentially ignore the sentence length, word alignment relationship, and other token-level information. Secondly, regularizing sentence representation mismatches to the working paradigm of the NMT model, because the decoder directly performs cross attention on the whole state sequences rather than the sentence representation. Besides,

[*] Corresponding author: Yang Feng.
Reproducible code: https://github.com/ictnlp/Zero-MNMT.

Findings of the Association for Computational Linguistics: EMNLP 2022, pages 6492–6504
December 7-11, 2022 ©2022 Association for Computational Linguistics

all the above work focuses on the encoder side and cannot help learn the universal mapping relationship for the decoder.

Given the above, we propose a method to learn the universal representations and cross-mappings to improve the zero-shot translation performance. Based on the optimal transport theory, we propose state mover's distance (SMD) to model the differences of two state sequences at the token level. To map the semantic-equivalent sentences from different languages to the same place of the semantic space, we add an auxiliary loss to minimize the SMD of the source and target sentences. Besides, we propose an agreement-based training scheme to learn universal mapping relationships for the translation directions with the same target language. We mixup the source and target sentences to obtain a pseudo sentence. Then, the decoder makes predictions separately conditioned on this pseudo sentence and the corresponding source or target sentences. We try to improve the prediction consistency by minimizing the KL divergence of the two output distributions. The experimental results on diverse multilingual datasets show that our method can bring 2~3 BLEU improvements over the strong baseline system and consistently outperform other contrast methods. The analysis proves that our method can better align the semantic space and improve the prediction consistency.

2 Background

In this section, we will give a brief introduction to the TRANSFORMER (Vaswani et al., 2017) model and the many-to-many multilingual translation.

2.1 The transformer

We denote the input sequence of symbols as $\mathbf{x} = (x_1, \ldots, x_{nx})$ and the ground-truth sequence as $\mathbf{y} = (y_1, \ldots, y_{ny})$. The transformer model is based on the encoder-decoder architecture. The encoder is composed of N identical layers. Each layer has two sublayers. The first is a multi-head self-attention sublayer, and the second is a fully connected feed-forward network. Both of the sublayers are followed by a residual connection operation and a layer normalization operation. The input sequence $\mathbf{x}$ will be first converted to a sequence of vectors. Then, this sequence of vectors will be fed into the encoder, and the output of the N-th layer will be taken as source state sequences. We denote it as $\mathbf{H_x}$. The decoder is also composed of N identical layers. In addition to the same kind of two sublayers in each encoder layer, the cross-attention sublayer is inserted between them, which performs multi-head attention over the output of the encoder. We can get the predicted probability of the k-th target word conditioned by the source sentence and the $k-1$ previous target words. The model is optimized by minimizing a cross-entropy loss of the ground-truth sequence with teacher forcing:

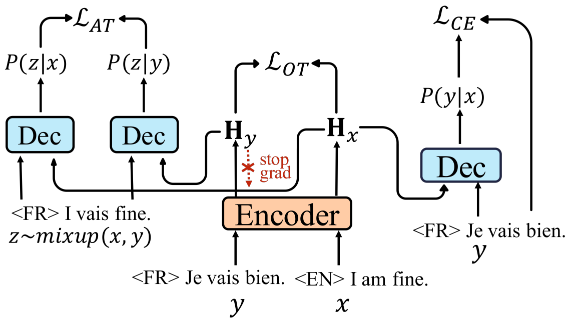


Figure 1: The training scheme of our method. $\mathbf{x}$ and $\mathbf{y}$ denote a pair of translations; $\mathbf{H_x}$ and $\mathbf{H_y}$ denote the corresponding state sequences. $\mathbf{z}$ is the pseudo sentence by mixuping $\mathbf{x}$ and $\mathbf{y}$. 'Dec' denotes the decoder and there is only one decoder in the model. 'stop-grad' denotes the stop-gradient operation during back propagation. $\mathcal{L}_{CE}$, $\mathcal{L}_{OT}$, and $\mathcal{L}_{AT}$ denote the cross entropy loss, optimal transport loss, and agreement-based training loss.

$$\mathcal{L}_{CE} = -\frac{1}{n_y}\sum_{k=1}^{n_y} \log p(y_k|\mathbf{y}_{<k}, \mathbf{x}; \theta), \quad (1)$$

where n_y is the length of the target sentence and θ denotes the model parameters.

2.2 Multilingual Translation

We define $L = \{l_1, \ldots, l_M\}$ where L is a collection of M languages involved in the training phase. Following Johnson et al. (2017), we share all the model parameters for all the languages. Following Liu et al. (2020), we add a particular language id token at the beginning of the source and target sentences, respectively, to indicate the language.

3 Method

The main idea of our method is to help the encoder output universal representations for all the languages and help the decoder map the semantic-equivalent representation from different languages to the target language's space. We propose two approaches to fulfill this goal. The first is to directly bridge the gap between the state sequences that carry the same semantics. The second is to force the decoder to make consistent predictions based on the semantic-equivalent sentences. Figure 1 shows the overall training scheme.

3.1 Optimal Transport

Earth Mover's Distance Based on the optimal transport theory (Villani, 2009; Peyré et al., 2019), the earth mover's distance (EMD) measures the minimum cost to transport the probability mass from one distribution to another distribution. Assuming that there are two probability distributions μ and μ', that are defined as:

$$\begin{aligned} \mu &= \{(\mathbf{w}_i, m_i)\}_{i=1}^{n}, \quad s.t. \sum_i m_i = 1; \\ \mu' &= \{(\mathbf{w}'_j, m'_j)\}_{j=1}^{n'}, \quad s.t. \sum_j m'_j = 1, \end{aligned} \tag{2}$$

where each data point $\mathbf{w}_i \in \mathbb{R}^d$ has a probability mass m_i ($m_i > 0$). There are n data points in μ. We define a cost function $c(\mathbf{w}_i, \mathbf{w}'_j)$ that determines the cost of per unit between two points $\mathbf{w}_i$ and $\mathbf{w}'_i$. Given above, the EMD is defined as:

$$\begin{aligned} \mathcal{D}(\mu, \mu') &= \min_{\mathbf{T}\geq 0} \sum_{i,j} \mathbf{T}_{ij} c(\mathbf{w}_i, \mathbf{w'}_j) \\ s.t. \quad & \sum_{j=1}^{n'} \mathbf{T}_{ij} = m_i, \forall i \in \{1, \ldots, n\}, \\ & \sum_{i=1}^{n} \mathbf{T}_{ij} = m'_j, \forall j \in \{1, \ldots, n'\}. \end{aligned} \tag{3}$$

$\mathbf{T}_{ij}$ denotes the mass transported from μ to μ'.

State Mover's Distance Following EMD, we define the state mover's distance (SMD) to measure the minimum 'travel cost' between two state sequences. Given a pair of translations $\mathbf{x} = (x_1, \ldots, x_{nx})$, and $\mathbf{y} = (y_1, \ldots, y_{ny})$, we can get their corresponding state sequences after feeding them to the encoder, which are denoted as:

$$\begin{aligned} \mathbf{H}_\mathbf{x} &= (\mathbf{h}_1, \ldots, \mathbf{h}_i, \ldots, \mathbf{h}_{nx}), \\ \mathbf{H}_\mathbf{y} &= (\mathbf{h}'_1, \ldots, \mathbf{h}'_j, \ldots, \mathbf{h}'_{ny}), \end{aligned} \tag{4}$$

where nx and ny denote the sentence length of the source and target sentences. We can regard $\mathbf{H}_\mathbf{x}$ as a discrete distribution on the space $\mathbb{R}^d$, where the probability only occurs at each specific point $\mathbf{h}_i$. Next, several previous studies (Schakel and Wilson, 2015; Yokoi et al., 2020) have confirmed that the embedding norm is related to the word importance, and the important words have larger norms. Inspired by these findings, we also observe that the state vector has similar properties. The state vectors of essential words, such as content and medium-frequency words, have larger norms than unimportant ones, such as function words, high-frequency words. Therefore, we propose to use the normalized vector norm as the probability mass for each state point:

$$m_i = \frac{|\mathbf{h}_i|}{\sum_i |\mathbf{h}_i|}, m'_j = \frac{|\mathbf{h}'_j|}{\sum_j |\mathbf{h}'_j|}, \tag{5}$$

where $|\cdot|$ denotes the norm of the vector.

Given above, we can convert the state sequences to distributions:

$$\begin{aligned} \mu_\mathbf{x}^\mathbf{H} &= \{(\mathbf{h}_i, \frac{|\mathbf{h}_i|}{\sum_i |\mathbf{h}_i|})\}_{i=1}^{nx}, \\ \mu_\mathbf{y}^\mathbf{H} &= \{(\mathbf{h}'_j, \frac{|\mathbf{h}'_j|}{\sum_j |\mathbf{h}'_j|})\}_{j=1}^{ny}. \end{aligned} \tag{6}$$

Then, the SMD is formally defined as follows:

$$\begin{aligned} \mathcal{D}(\mu_\mathbf{x}^\mathbf{H}, \mu_\mathbf{y}^\mathbf{H}) &= \min_{\mathbf{T}\geq 0} \sum_{i,j} \mathbf{T}_{ij} c(\mathbf{h}_i, \mathbf{h'}_j), \\ s.t. \quad & \sum_{j=1}^{ny} \mathbf{T}_{ij} = \frac{|\mathbf{h}_i|}{\sum_i |\mathbf{h}_i|}, \forall i \in \{1, \ldots, nx\}, \\ & \sum_{i=1}^{nx} \mathbf{T}_{ij} = \frac{|\mathbf{h}'_j|}{\sum_j |\mathbf{h}'_j|}, \forall j \in \{1, \ldots, ny\}. \end{aligned} \tag{7}$$

As illustrated before, we want decoder to make consistent predictions conditioned on the equivalent state sequences. Considering that the vector norm and direction both have impacts on the cross-attention results of decoder, we use the Euclidean distance as the cost function. We didn't use the cosine similarity based metric, because it only considers the impact of vector direction. The proposed SMD is a fully unsupervised algorithm to align the contextual representations of the two semantic-equivalent sentences.

Approximation of SMD The exact computation to SMD is a linear programming problem with typical super $O(n^3)$ complexity, which will slow down the training speed greatly. We can obtain a relaxed bound of SMD by removing one of the two constraints, respectively. Following Kusner et al.

(2015), we remove the second constraints:

$$\mathcal{D}^*(\mu_{\mathbf{x}}^{\mathbf{H}}, \mu_{\mathbf{y}}^{\mathbf{H}}) = \min_{\mathbf{T}\geq 0} \sum_{i,j} \mathbf{T}_{ij} c(\mathbf{h}_i, \mathbf{h'}_j),$$
$$s.t. \quad \sum_{j=1}^{ny} \mathbf{T}_{ij} = \frac{|\mathbf{h}_i|}{\sum_i |\mathbf{h}_i|}, \forall i \in \{1, \ldots, nx\}. \tag{8}$$

The above approximation must yield a lower bound to the exact SMD distance. The accurate SMD solution that satisfies both of the two constraints must also satisfy the first constraint. Given the approximation, the optimal solution for each state vector $\mathbf{h}_i$ is to move all its probability mass to the most similar state vector $\mathbf{h}'_j$. Therefore, the approximation also enables the many-to-one alignment relationships during training. We have also tried some approximation algorithms that can get a more accurate estimation of SMD, e.g., Sinkhorn algorithm(Cuturi, 2013), IPOT (Xie et al., 2020). However, we have not observed consistent improvements in our preliminary experiments, and these algorithms also slow down the training speed significantly.

Objective Function We define a symmetrical loss to minimize the SMD of both sides:

$$\mathcal{L}_{OT} = \frac{1}{2}\left(\mathcal{D}^*(\mu_{\mathbf{x}}^{\mathbf{H}}, \mu_{\mathbf{y}}^{\mathbf{H}}) + \mathcal{D}^*(\mu_{\mathbf{y}}^{\mathbf{H}}, \mu_{\mathbf{x}}^{\mathbf{H}})\right). \tag{9}$$

3.2 Agreement-based Training

Theoretical Analysis In zero-shot translation, the decoder should map the semantic representations from different languages to the target language space, even if it has never seen the translation directions during training. This ability needs the model to make consistent predictions based on the semantic-equivalent sentences, whatever the input language is. To improve the prediction consistency of the model, we propose an agreement-based training method. Because the source sentence $\mathbf{x}$ and target sentence $\mathbf{y}$ are semantically equivalent, the probability of predicting any other sentence $\mathbf{z}$ based on them should be always equal theoretically, which is denoted as:

$$p(\mathbf{z}|\mathbf{x}) = p(\mathbf{z}|\mathbf{y}). \tag{10}$$

Specifically, the predicted probabilities of the k-th target word conditioned by the first $k-1$ words of $\mathbf{z}$ and the source and target sentences is equal:

$$p(z_k|\mathbf{z}_{<k}, \mathbf{x}; \theta) = p(z_k|\mathbf{z}_{<k}, \mathbf{y}; \theta), \tag{11}$$

where θ denotes the model parameters. Optimizing Equation 11 can not only help the encoder produce universal semantic representations but also help the decoder map different source languages to the particular target language space indicated by $\mathbf{z}$.

Mixup for z Although Equation 11 is theoretically attractive, the choice of sentence $\mathbf{z}$ has a significant influence on the above optimization. If we use a random sentence as $\mathbf{z}$, which is not related to $\mathbf{x}$ and $\mathbf{y}$, the prediction makes no sense, and the model learns helpful nothing. If we use either $\mathbf{x}$ or $\mathbf{y}$ directly, this will cause information leakage on one side of Equation 11. As a result, the prediction difficulty between the two sides differs significantly, and it is hard for one side to catch up with the other side. Given the above, we need a inter-sentence that is "between" $\mathbf{x}$ and $\mathbf{y}$. Inspired by the success of mixup technique in NLP (Zhang et al., 2020b; Cheng et al., 2021), we generate a pseudo sentence by hard mixuping $\mathbf{x}$ and $\mathbf{y}$ at token-level. We truncate the longer sentences of $\mathbf{x}$ and $\mathbf{y}$ to make them equal in length. Since these two sentences are translation pairs, their sentence lengths are usually close, truncating will not significantly reduce the length of the longer sentence and will not enhance the decoder learn shorter outputs. We denote the truncated sentence as $\mathbf{x}'$ and $\mathbf{y}'$, and their length as n'. Then we can generate $\mathbf{z}$ as:

$$\mathbf{z} = \mathbf{g} \odot \mathbf{x}' + (1 - \mathbf{g}) \odot \mathbf{y}', \tag{12}$$

where $\mathbf{g} \in \{0, 1\}^{n'}$, $\odot$ denotes the element-wise product. Each element in $\mathbf{g}$ is sampled from Bernoulli(λ), where the parameter λ is sampled from Beta(α, β), and α and β are two hyperparameters. The language tag in $\mathbf{z}$, which determines the translation direction, is either come from $\mathbf{x}$ or $\mathbf{y}$.

Objective Function Similar to Equation 9, we define another symmetrical loss based on the KL divergence of the model prediction distributions:

$$\begin{aligned}\mathcal{L}_{AT} = &\frac{1}{2n'}\sum_{k=1}^{n'} KL\left(p(z_k|\mathbf{z}_{<k}, \mathbf{H}_{\mathbf{x}})||p(z_k|\mathbf{z}_{<k}, \mathbf{H}_{\mathbf{y}})\right)\\ &+ KL\left(p(z_k|\mathbf{z}_{<k}, \mathbf{H}_{\mathbf{y}})||p(z_k|\mathbf{z}_{<k}, \mathbf{H}_{\mathbf{x}})\right).\end{aligned} \tag{13}$$

We omit the model parameters for convenience.

3.3 The Final Loss

The final loss consists of three parts, the cross entropy loss (Equation 1), the optimal transport loss based on SMD (Equation 9) and the KL divergence

Dataset	Language Pairs	Size
IWSLT	En↔{De, It, Nl, Ro}	1.79M
IWSLT-b	Nl↔De↔En↔It↔Ro	1.79M
PC-6	En↔{Kk, Tr, Ro, Cs, Ru}	7.9M
OPUS-7	En↔{De, Fr, Nl, Ru, Zh, Ar}	11.6M

Table 1: The statics of our datasets.

loss for the agreement-based training (Equation 13):

$$\mathcal{L} = \mathcal{L}_{CE} + \gamma_1|\mathbf{x}|\mathcal{L}_{OT} + \gamma_2\mathcal{L}_{AT} \tag{14}$$

where γ_1 and γ_2 are two hyperparameters that control the contributions of the two regularization loss terms. Since $\mathcal{L}_{OT}$ is calculated on the sentence-level and the other two losses are calculated on the token-level, we multiply the averaged sequence length $|\mathbf{x}|$ to $\mathcal{L}_{OT}$. Among these three losses, the first term dominates the parameter update of the model, and determines the model performance mostly. The latter two regularization loss terms only slightly modify the directions of the gradients. Because the first loss term does not depend on $\mathbf{H_y}$, we apply the stop-gradient operation to $\mathbf{H_y}$ (Figure 1), which means that the gradients will not pass through $\mathbf{H_y}$ to the encoder.

4 Experiments

4.1 Data Preparation

We conduct experiments on the following multilingual datasets: IWSLT17, PC-6, and OPUS-7. The brief statistics of the training set are in Table 1. We put more details in the appendix.

IWSLT17 (Cettolo et al., 2017) We simulate two scenarios. The first (IWSLT) is English-pivot, where we only retain the parallel sentences from/to English. The second (IWSLT-b) has a chain of pivots, where two languages are connected by a chain of pivot languages. Each translation direction has about 0.22M sentence pairs. Both of the two scenarios have eight supervised translation directions and twelve zero-shot translation directions. We use the official validation and test sets.

PC-6 The PC-6 dataset is extracted from the PC-32 corpus (Lin et al., 2020). The data amount of different language pairs is unbalanced, ranging from 0.12M to 1.84M. This dataset has ten supervised and twenty zero-shot translation directions. We use the validation and test sets collected from WMT16~19 for the supervised directions. The zero-shot validation and test sets are extracted from the WikiMatrix (Schwenk et al., 2021), each containing about 1K~2K sentences pairs.

OPUS-7 The OPUS-7 dataset is extracted from the OPUS-100 corpus (Zhang et al., 2020a). The language pairs come from different language families and have significant differences. This dataset has twelve supervised translation directions and thirty zero-shot translation directions. We use the standard validation and test sets released by Zhang et al. (2020a). We concatenate the zero-shot test sets with the same target language for convenience.

We use the Stanford word segmenter (Tseng et al., 2005; Monroe et al., 2014) to segment Arabic and Chinese, and the Moses toolkit (Koehn et al., 2007) to tokenize other languages. Besides, integrating operations of 32K is performed to learn BPE (Sennrich et al., 2016).

4.2 Systems

We use the open-source toolkit called *Fairseq-py* (Ott et al., 2019) as our Transformer system. We implement the following systems:

- **Zero-Shot (ZS)** The baseline system which is trained only with the cross-entropy loss (Equation 1). Then the model is tested directly on the zero-shot test sets.
- **Pivot Translation (PivT)** (Cheng et al., 2017) The same translation model as ZS. The model first translates the source language to the pivot language and then generates the target language.
- **Sentence Representation Alignment (SRA)** (Arivazhagan et al., 2019) This methods adds an regularization loss to minimize the discrepancy of the source and target sentence representations.

$$\mathcal{L} = \mathcal{L}_{CE} + \gamma Dis(Enc(s), Enc(t)), \tag{15}$$

where 'Dis' denotes the distance function and 'Enc(·)' denotes the sentence representations. We use the averaged sentence representation and Euclidean distance function because we find they work better. We vary the hyperparameter γ from 0.1 to 1 to tune the performance.

- **Softmax Forcing (SF)** (Pham et al., 2019) This method enable the decoder to generate the target sentence from itself by adding an extra loss:

$$\mathcal{L}_{SF} = \gamma \sum_{k}^{n_y} KL(p(y_k|\mathbf{y}_{<k}, \mathbf{x})||p(y_k|\mathbf{y}_{<k}, \mathbf{y})) \tag{16}$$

The γ is tuned as in the 'SRA' system.

IWSLT	De-It		De-Nl		De-Ro		It-Ro		It-Nl		Nl-Ro		Zero	Sup.
Model	→	←	→	←	→	←	→	←	→	←	→	←	Avg.	Avg
ZS	15.64	15.28	18.46	18.14	14.42	14.98	17.91	20.14	18.16	18.79	15.81	16.41	17.01	30.62
SRA	16.44	16.45	18.44	19.15	15.07	15.83	18.52	21.52	19.3	19.1	16.83	17.66	17.85	30.41
SF	16.34	15.77	18.37	18.16	14.74	15.25	18.54	21.64	18.6	19.18	16.09	16.94	17.46	30.5
CL	17.37	16.58	19.69	19.5	15.51	16.25	18.91	22.58	18.78	20.02	17.27	17.91	18.36	30.39
DisPos	16.62	15.64	19.64	18.78	15.07	15.96	18.67	21.56	19.01	20.15	16.46	18.18	17.97	30.49
DT	16.82	15.81	18.74	18.64	15.12	16.32	18.70	22.13	18.92	19.29	16.21	18.22	17.91	30.51
TGP	16.77	**18.51**	14.58	17.12	**16.84**	**16.88**	19.42	19.25	20.01	19.04	21.67	18.43	18.21	**30.66**
LMP	16.87	18.44	15.05	16.66	16.20	16.12	19.04	19.05	19.35	18.68	**22.17**	17.97	17.96	30.52
PivT	18.31	17.9	19.99	19.33	15.54	17.45	19.77	22.97	21.43	21.44	17.57	19.82	19.29	-
ZS+OT	17.35	17.08	19.77	19.05	15.66	16.17	**19.71**	22.32	**20.18**	**20.57**	16.87	18.09	18.56	30.42
ZS+AT	16.37	15.84	19.11	18.41	14.85	15.59	18.37	21.09	18.77	19.4	15.86	17.46	17.59	30.55
Ours	**17.53**	17.03	**19.94**	**19.67**	15.61	16.57	19.23	**22.42**	20.05	20.23	**17.05**	**18.64**	**18.66**	30.52

IWSLT-b	De-It		En-Nl		De-Ro		En-Ro		It-Nl		Nl-Ro		Zero	Sup.
Model	→	←	→	←	→	←	→	←	→	←	→	←	Avg.	Avg.
ZS	17.79	17.3	25.48	30.99	15.65	17.28	21.7	30.14	20.79	21.02	15.74	17.28	20.93	**30.46**
SRA	18.09	18.05	26.52	31.15	15.8	17.43	22.24	30.19	20.35	20.65	16.39	17.83	21.22	30.29
SF	18.25	17.61	26	31.28	16.06	17.51	22.43	30.51	20.67	20.82	16.2	17.24	21.21	30.35
CL	**18.49**	18.29	26.88	31.46	15.71	17.23	23.01	30.78	20.62	20.8	16.58	18.17	21.5	30.28
DisPos	17.98	17.35	26.26	31.13	15.75	**18.07**	22.95	30.45	**21.02**	20.58	16.38	18.28	21.35	29.89
TGP	18.22	18.69	26.62	30.96	15.57	17.26	23.21	30.22	20.62	20.38	16.58	17.65	21.33	30.33
LMP	18.36	**18.83**	27.2	30.5	16.05	17.05	**23.99**	29.38	20.57	19.83	16.72	17.56	21.33	30.37
PivT	18.38	19.08	27.3	28.02	15	16.35	23.72	28.72	20.34	19.45	15.7	16.8	20.74	-
ZS+OT	18.09	18.06	26.6	**31.69**	15.76	17.19	23.46	**30.99**	20.31	**20.86**	16.92	18.05	21.49	30.37
ZS+AT	18.23	17.51	26.24	31.12	**16.19**	17.5	22.64	30.33	20.72	20.59	16.29	17.64	21.25	30.39
Ours	18.41	18.05	**27.39**	31.36	16.15	17.48	23.22	30.9	20.68	20.82	**17.03**	**18.29**	**21.64**	30.33

PC-6	x→Kk	x→Tr	x→Ro	x→Cs	x→Ru	Zero Avg.	Sup. Avg.	OPUS-7	x→De	x→Fr	x→Nl	x→Ru	x→Zh	x→Ar	Zero Avg.	Sup. Avg.
ZS	5.87	9.29	14.23	13.55	16.83	11.95	**21.73**	ZS	13.58	22.63	17.96	15.42	29.78	21.58	20.15	**34.2**
SRA	5.90	10.09	17.36	15.85	19.31	13.68	21.66	SRA	17.04	26.12	19.29	20.9	31.99	22.01	22.89	33.97
SF	4.76	9.95	17.77	15.83	20.10	13.68	21.64	SF	15.99	25.2	18.2	20.85	31.65	21.5	22.23	33.99
CL	6.07	10.72	17.96	16.14	21.58	14.49	21.54	CL	17.41	26.19	19.66	21.1	32.52	21.69	23.09	33.86
DisPos	6.60	10.14	15.47	15.89	18.70	12.51	21.45	DisPos	15.95	25.36	18.86	19.75	31.34	22.08	22.22	34.12
DT	6.92	10.49	17.37	15.63	21.74	14.43	21.61	DT	14.97	23.95	18.10	18.91	29.65	20.68	21.04	34.03
TGP	**7.33**	10.98	**20.63**	13.81	21.21	14.79	21.58	TGP	16.86	25.65	18.99	20.83	32.47	21.47	22.71	34.18
LMP	4.45	8.50	16.42	15.25	19.28	12.78	21.71	LMP	14.65	23.94	18.36	19.02	30.58	20.99	21.26	34.07
PivT	4.29	10.59	19.23	17.22	21.65	14.58	-	PivT	17.97	28.37	19.76	22.97	34.08	23.74	24.48	-
ZS+OT	6.22	11.08	18.74	16.86	22.61	15.1	21.6	ZS+OT	17.56	26.70	19.54	21.88	32.42	22.48	23.43	34.02
ZS+AT	6.04	10.74	17.92	15.69	20.63	14.2	21.72	ZS+AT	16.78	25.89	18.93	21.21	32.02	21.72	22.75	34.1
Ours	6.58	**11.44**	18.55	**17.11**	**22.77**	**15.29**	21.68	Ours	**17.60**	**26.74**	**19.68**	**21.91**	**32.63**	**23.24**	**23.63**	34.17

Table 2: The overall BLEU scores on the test sets. "Zero Avg." and "Sup. Avg." denote the average BLEU scores on the zero-shot and supervised directions. The "x" in the third table denotes all languages except for the target language. The highest scores are marked in bold for all models except for the "PivT" system in each column.

- **Contrastive Learning (CL)** (Pan et al., 2021) This method adds an extra contrastive loss to minimize the representation gap of similar sentences and maximize that of irrelevant sentences:

$$\mathcal{L}_{CL} = -\gamma \log \frac{e^{sim^{+}(\mathcal{R}(s),\mathcal{R}(t))/\tau}}{\sum_{w} e^{sim^{-}(\mathcal{R}(s),\mathcal{R}(w))/\tau}}, \quad (17)$$

where $+$ and $-$ denote positive and negative sample pairs, $\mathcal{R}(\cdot)$ denotes the averaged state representations. We set τ as 0.1 as suggested in the paper and tune γ as in the 'SRA' system.

- **Disentangling Positional Information (DisPos)** (Liu et al., 2021) This method removes the residual connections in a middle layer of the encoder to get the language-agnostic representations.
- **Denosing Training (DT)** (Wang et al., 2021) This method introduces a denoising auto-encoder objective during training.
- **Target Gradient Projection (TGP)** (Yang et al., 2021b) This method projects the training gradient to not conflict with the oracle gradient of a small amount of direct data.
- **Language Model Pre-training (LMP)** (Gu et al., 2019) This method strengthens the decoder language model prior to machine translation training.

The following systems are implemented based on our method:

- **ZS+OT** We only add the optimal transport loss

(Equation 9) during training. We vary the hyperparameter γ_1 from 0.1 to 1, and we find that it can constantly improve the performance whatever γ_1 is. The detailed results and the final setting about the hyperparameter are put in the appendix.

• **ZS+AT** We only add the agreement-based training loss (Equation 13) during training. The α and β in the beta distribution are set as 6 and 3, respectively. We vary the hyperparameter γ_2 from 10^{-4} to 0.1.

• **ZS+OT+AT (Ours)** The model is trained with the complete objective function (Equation 14). The hyperparameters are set according to the searched results of the above two systems and are listed in the appendix.

Implementation Details All the systems are implemented as the base model configuration in Vaswani et al. (2017) strictly. We employ the Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.98$. We use the inverse square root learning scheduler and set the $warmup\_steps = 4000$ and $lr = 0.0007$. We set dropout as 0.3 for the IWSLT datasets and 0.1 for the for the PC-6 and OPUS-7 datasets. All the systems are trained on 4 RTX3090 GPUs with the update frequency 2. The max token is 4096 for each GPU. For the IWSLT data sets, we first pretrain the model with the cross-entropy loss (Equation 1) for 20K steps and then continually train the model combined with the proposed loss terms for 80K steps. For the PC-6 and OPUS-7 datasets, the pretraining steps and continual-training steps are both 100k.

4.3 Main Results

All the results (including the intermediate results of the 'PivT' system) are generated with beam size = 5 and length penalty $\alpha = 0.6$. The translation quality is evaluated using the case-sensitive BLEU (Papineni et al., 2002) with the *SacreBLEU* tool (Post, 2018). We report the tokenized BLEU for Arabic, char-based BLEU for Chinese, and detokenized BLEU for other languages[1]. The main results are shown in Table 2. We report the averaged BLEU with the same target language on the PC-6 and OPUS-7 dataset for display convenience, and the detailed results are in the appendix. The 'Ours' system significantly improves over the 'ZS' baseline system and outperforms other zero-shot-based systems on all datasets. The two proposed methods, OT and AT, can both help the model learn universal and cross mappings , so they both can improve the model performance independently. These two methods also complement each other and can further improve the performance when combined together. Besides, 'Ours' system can even exceed the 'PivT' system when the distant language pairs in the IWSLT-b or the low-resource language pairs in the PC-6 bring severe error accumulation problems. We also compare the training speed and put the results in the appendix.

IWSLT	x-De	x-It	x-Nl	Avg.
ZS	21.5	20.79	19.99	20.76
SRA	21.79	21.92	20.67	21.46
CL	23.47	21.52	21.09	22.03
Ours	**23.6**	**23.33**	**21.48**	**22.80**

Table 3: The pair-wise BLEU on the IWSLT three-way-parallel test sets.

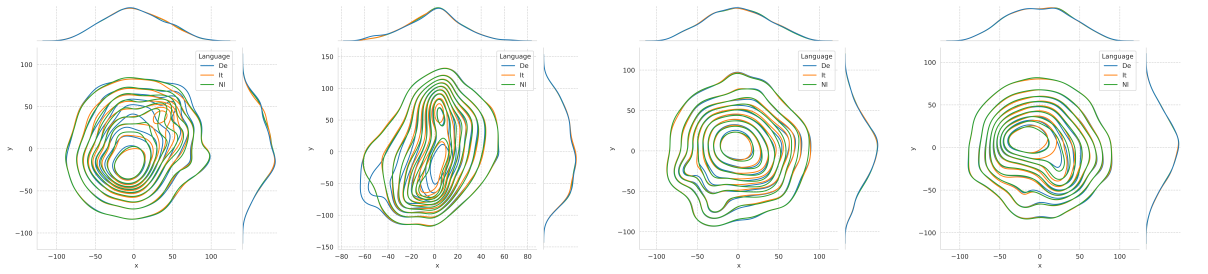
5 Analysis

In this section, we try to understand how our method improves the zero-shot translation.

5.1 Sentence Representation Visualization

To verify whether our method can better align different languages' semantic space, we visualize each model's encoder output with the IWSLT test sets. We first select three languages: Germany, Italian, and Dutch. Then we filter out the overlapped sentences of the three languages from the corresponding test sets and create a new three-way-parallel test set. Next, we feed all the sentences to the encoder of each model and average the encoder output to get the sentence representation. Last, we apply dimension reduction to the representation with t-SNE (Van der Maaten and Hinton, 2008). The visualization result in Figure 2(a) shows that the 'ZS' system cannot align the three languages well, which partly confirms our assumption that the conventional MNMT cannot learn universal representations for all languages. As a contrast, the 'Ours' system (d) can draw the representation closer and achieve comparative results as the 'CL' system (c) without large amounts of negative instances to contrast. The visualization results confirm that our method can learn good universal representation for different languages.

[1] BLEU+case.mixed+numrefs.1+smooth.exp+tok.{13a,none,zh}+version.1.5.1



(a) ZS (b) SRA (c) CL (d) Ours

Figure 2: The visualization of sentence representation after dimension reduction on the IWSLT three-way-parallel test sets. The blue line denotes Germany, the orange line denotes Italian, and the green line denotes Dutch.

System	IWSLT	IWSLT-b	PC-6	OPUS-7
ZS	93.2%	93.72%	87.93%	74.1%
SRA	93.9%	93.88%	91.54%	85.83%
CL	93.97%	93.96%	91.76%	86.23%
Ours	**94.03%**	**94.06%**	**93.24%**	**86.75%**

Table 4: The target language prediction accuracy.

5.2 Inspecting Prediction Consistency

To verify whether our method can help map the semantic representation from different languages to the same space of the target language, we inspect the prediction consistency of the models when the model is fed with synonymous sentences from different languages. Precisely, we measure the pair-wise BLEU on the above IWSLT three-way-parallel test set. We choose one language as the target language, e.g., German, and then translate the other two languages, e.g., Italian and Dutch, to the target language. After obtaining these two translation files, we use one file as the reference, the other as the translation to calculate the BLEU, and then we swap the role of these two files to calculate the BLEU again. We average the BLEU scores to get the pair-wise BLEU, and the results in Table 3 show that our method can achieve higher results, which proves that our method can improve the prediction consistency.

5.3 Inspecting Spurious Correlations

The zero-shot translation usually suffers from capturing spurious correlations in the supervised directions, which means that the model overfits the mapping relationship from the input language to the output language observed in the training set (Gu et al., 2019). This problem often causes the off-target prediction phenomenon where the model generates translation in the wrong target languages. To check whether our method can alleviate this phenomenon, we use the Langdetect [2] toolkit to identify the target language and calculate the prediction accuracy as $1 - n_{off-target}/n_{total}$. We also compare our method with the 'SRA' and 'CL' methods. The results are shown in Table 4. The 'ZS' baseline system can achieve high prediction accuracy on the IWSLT dataset, but the performance begin to decline as the amount of data becomes unbalanced and the languages become more unrelated. On all the datasets, our method achieves higher prediction accuracy and outperforms all the contrast methods. We can conclude from the results that our method can reduce the spurious correlation captured by the model.

6 Related Wrok

Recent work on zero-shot translation can be divided into two categories. The first category helps the encoder produce language-agnostic features via extra regularization loss or training tasks. Pham et al. (2019) propose to compress the output of the encoder into a consistent number of states. Arivazhagan et al. (2019) maximize the cosine similarities between the averaged representations of the source and target sentences. Pan et al. (2021) and Wei et al. (2021) propose contrastive learning schemes to minimize the averaged sentence representation gap of similar sentences and maximize that of irrelevant sentences. Compared with their methods, we directly bridge the gap between two state sequences, which alleviates the mismatch problem of sentence representation. Ji et al. (2020) leverage explicit alignment information by external aligner tool or additional attention layer to obtain the aligned words for masking, and then they let the model predict the masked words based on the

[2] https://github.com/Mimino666/langdetect

surrounding words. Compared with this work, our method is to align the whole state sequences of different languages, not just for single words. Liu et al. (2021) remove the residual connections in a middle layer of the encoder to release the positional correspondence to input tokens. Wang et al. (2021) introduce a denoising auto-encoder objective to improve the translation accuracy. Yang et al. (2021b) leverage an auxiliary target language prediction task to retain information about the target languages. Z. et al. (2022) uses optimal transport theory to improve the low-resource neural machine translation. Compared with these work, our method introduces explicit constraints to the semantic representations.

The second category extends the training data by generating pseudo sentence pairs or utilizing monolingual data. Gu et al. (2019) apply decoder pre-training and back-translation to improve the zero-shot ability. Al-Shedivat and Parikh (2019) first translate the source and target languages to a third language and then make consistent predictions based on this pseudo sentence. Zhang et al. (2020a) propose random online back translation to enforce the translation of unseen training language pairs. Chen et al. (2021) fuse the pretrained multilingual model to the NMT model. Compared with these works, our method does not need additional data or additional time to generate pseudo corpus. If necessary, our method can also be combined with these works to further improve the zero-shot performance of the model. Yang et al. (2021a) propose to substitute some fragments of the source language with their counterpart translations to get the code-switch sentences. Compared to this work, our agreement-based method mixups the translation pairs to generate the pseudo sentence as the decoder input and then help the model to make consistent predictions.

7 Conclusion

In this work, we focus on improving the zero-shot ability of multilingual neural machine translation. To reduce the discrepancy of the encoder output, we propose the state mover's distance based on the optimal transport theory and directly minimize the distance during training. We also propose an agreement-based training method to help the decoder make consistent predictions based on the semantic-equivalent sentences. The experimental results show that our method can get consistent improvements on diverse multilingual datasets. Further analysis shows that our method can better align the semantic space, improve the prediction consistency, and reduce the spurious correlations.

Limitations

Although our method can improve the performance of the zero-shot translation directions, it has limited benefits for the supervised translation performance. On the one hand, the vanilla MNMT model has already been able to learn a lot of language shared knowledge. On the other hand, the language-specific knowledge learned by the model can also help the model achieve good translation performance in the supervised translation directions. Therefore, our method is limited to improving the supervised translation performance. Besides, some reviewers pointed out that our method degraded the supervised translation performance according to the results of the main experiments. This is because we select the checkpoints based on the performance of the zero-shot valid sets, which may cause a slight decline in the performance of the supervised directions. If we select checkpoints based on the the supervised valid sets, our method can improve the zero-shot performance without degrading the BLEU of the supervised directions.

Acknowledgements

We thank all the anonymous reviewers for their insightful and valuable comments.

References

Maruan Al-Shedivat and Ankur P. Parikh. 2019. Consistency by agreement in zero-shot neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 1184–1197.

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

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.

Guanhua Chen, Shuming Ma, Yun Chen, Li Dong, Dongdong Zhang, Jia Pan, Wenping Wang, and Furu Wei. 2021. Zero-shot cross-lingual transfer of neural machine translation with multilingual pretrained encoders. In *EMNLP 2021*.

Yong Cheng, Wei Wang, Lu Jiang, and Wolfgang Macherey. 2021. Self-supervised and supervised joint training for resource-rich machine translation. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, pages 1825–1835.

Yong Cheng, Qian Yang, Yang Liu, Maosong Sun, and Wei Xu. 2017. Joint training for pivot-based neural machine translation. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 3974–3980.

Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26:2292–2300.

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. *CoRR*, abs/2010.11125.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 866–875.

Jiatao Gu, Hany Hassan, Jacob Devlin, and Victor O. K. Li. 2018. Universal neural machine translation for extremely low resource languages. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 344–354.

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 Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1258–1268.

Thanh-Le Ha, Jan Niehues, and Alexander H. Waibel. 2016. Toward multilingual neural machine translation with universal encoder and decoder. *CoRR*, abs/1611.04798.

Baijun Ji, Zhirui Zhang, Xiangyu Duan, Min Zhang, Boxing Chen, and Weihua Luo. 2020. Cross-lingual pre-training based transfer for zero-shot neural machine translation. In *AAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 115–122.

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

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *ACL 2007, Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, June 23-30, 2007, Prague, Czech Republic*.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From word embeddings to document distances. In *International conference on machine learning*, pages 957–966. PMLR.

Zehui Lin, Xiao Pan, Mingxuan Wang, Xipeng Qiu, Jiangtao Feng, Hao Zhou, and Lei Li. 2020. Pretraining multilingual neural machine translation by leveraging alignment information. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 2649–2663.

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, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 1259–1273.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pretraining for neural machine translation. *Trans. Assoc. Comput. Linguistics*, 8:726–742.

Will Monroe, Spence Green, and Christopher D. Manning. 2014. Word segmentation of informal arabic with domain adaptation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 2: Short Papers*, pages 206–211.

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: Human Language Technologies, NAACL-HLT 2019, Minneapo-*

lis, MN, USA, June 2-7, 2019, Demonstrations, pages 48–53.

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, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 244–258.

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, July 6-12, 2002, Philadelphia, PA, USA*, pages 311–318.

Gabriel Peyré, Marco Cuturi, et al. 2019. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607.

Ngoc-Quan Pham, Jan Niehues, Thanh-Le Ha, and Alexander H. Waibel. 2019. Improving zero-shot translation with language-independent constraints. In *Proceedings of the Fourth Conference on Machine Translation, WMT 2019, Florence, Italy, August 1-2, 2019 - Volume 1: Research Papers*, pages 13–23.

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, Belgium, Brussels. Association for Computational Linguistics.

Adriaan M. J. Schakel and Benjamin J. Wilson. 2015. Measuring word significance using distributed representations of words. *CoRR*, abs/1508.02297.

Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2021. Wikimatrix: Mining 135m parallel sentences in 1620 language pairs from wikipedia. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 1351–1361.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Huihsin Tseng, Pi-Chuan Chang, Galen Andrew, Daniel Jurafsky, and Christopher D. Manning. 2005. A conditional random field word segmenter for sighan bakeoff 2005. In *Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing, SIGHAN@IJCNLP 2005, Jeju Island, Korea, 14-15, 2005*.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems*, pages 5998–6008.

Cédric Villani. 2009. *Optimal transport: old and new*, volume 338. Springer.

Weizhi Wang, Zhirui Zhang, Yichao Du, Boxing Chen, Jun Xie, and Weihua Luo. 2021. Rethinking zero-shot neural machine translation: From a perspective of latent variables. In *EMNLP 2021, Findings*.

Xiangpeng Wei, Rongxiang Weng, Yue Hu, Luxi Xing, Heng Yu, and Weihua Luo. 2021. On learning universal representations across languages. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*.

Yujia Xie, Xiangfeng Wang, Ruijia Wang, and Hongyuan Zha. 2020. A fast proximal point method for computing exact wasserstein distance. In *Uncertainty in Artificial Intelligence*, pages 433–453. PMLR.

Jian Yang, Yuwei Yin, Shuming Ma, Haoyang Huang, Dongdong Zhang, Zhoujun Li, and Furu Wei. 2021a. Multilingual agreement for multilingual neural machine translation. In *ACL/IJCNLP 2021, (Volume 2: Short Papers), Virtual Event, August 1-6, 2021*, pages 233–239.

Yilin Yang, Akiko Eriguchi, Alexandre Muzio, Prasad Tadepalli, Stefan Lee, and Hany Hassan. 2021b. Improving multilingual translation by representation and gradient regularization. In *EMNLP 2021, Long Paper*.

Sho Yokoi, Ryo Takahashi, Reina Akama, Jun Suzuki, and Kentaro Inui. 2020. Word rotator's distance. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 2944–2960.

Yang Z., Fang Q., and Y. Feng. 2022. Low-resource neural machine translation with cross-modal alignment. In *EMNLP 2022 Main Conference Long Paper*.

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, ACL 2020, Online, July 5-10, 2020*, pages 1628–1639.

Rongzhi Zhang, Yue Yu, and Chao Zhang. 2020b. Seqmix: Augmenting active sequence labeling via sequence mixup. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language*

Processing, EMNLP 2020, Online, November 16-20, 2020, pages 8566–8579.

A Appendix

A.1 PC-6 Data

The detailed statistics about the PC-6 corpus are shown in Table 6

A.2 Experiments Results on PC-6

The detailed results on the PC-6 corpus are shown in Table 5.

A.3 Hyperparameters

γ_1 **and** γ_2 The hyperparameter γ_1 and γ_2 in Equation 14 are set as in Table 7.

α **and** β We tried several combinations of α and β, and report the averaged BLEU in Table. Under the optimal setting ($\alpha = 6, \beta = 3$), the probability expectation that the words of the pseudo sentence $\mathbf{z}$ come from the source sentence $\mathbf{x}$ is 0.67 and from the target sentence $\mathbf{y}$ is 0.33.

A.4 Training Speed

We test the training speed of all the systems. All the speeds are measured as kilo-words per second (kwps) and tested in parallel on 4 RTX3090 GPUs with the same max token and update frequency. We also report the speed ratios of different systems compared with the speed of the ZS system. The results are shown in Table 9. The results show that our ’ZS+OT’ system is faster than the ’SRA’ and ’CL’ systems with better performance. The ’ZS+AT’ system is much slower because it needs three complete forward propagations.

PC-6	Cs-Kk		Kk-Ru		Ro-Ru		Tr-Ro		Cs-Ro		Cs-Ru	
	→	←	→	←	→	←	→	←	→	←	→	←
PivT	1.77	2.55	11.37	10.51	32.86	28.1	20.03	14.47	25.47	25.7	27.05	24.26
ZS	2.07	2.69	15.61	15.7	20.65	20.6	13.44	11.69	19.82	18.81	20.19	20.26
SRA	2.15	2.5	17.03	16.86	25.37	25.66	17.35	14.6	23.62	23.91	22.68	21.6
CL	1.99	2.68	16.48	16.49	29.28	26.8	17.82	15.66	23.87	23.42	27.05	24.29
DisPos	**2.24**	2.74	17.14	**17.95**	21.87	23.47	14.73	13.52	20.42	19.96	27.18	25.7
DT	2.2	2.87	19.23	18.88	28.05	25.88	17.82	14.41	22.29	22.3	26.29	23.98
TLP	2.01	2.82	14.59	13.01	28.41	25.88	18.53	13.25	23.11	22.54	25.24	22.74
ZS+OT	2.16	3.02	18.12	16.35	**30.71**	**27.84**	19.18	15.63	**24.44**	24.17	27.18	**25.71**
ZS+AT	2.06	2.82	15.8	16.54	28.01	26.37	19.25	15.59	22.63	22.55	24.6	23.27
Ours	2.2	**3.08**	**18.3**	17.91	30.59	27.73	**19.66**	**16.16**	23.58	**24.49**	**27.22**	25.66

PC-6	Cs-Tr		Kk-Ro		Kk-Tr		Ru-Tr		Zero
	→	←	→	←	→	←	→	←	Avg.
PivT	13.37	16.36	3.3	2.75	2.91	2.11	11.59	15.31	14.58
ZS	11	12.44	3.06	3.26	3.81	2.44	10.66	10.88	11.95
SRA	12.32	**15.37**	2.82	2.72	2.7	1.87	10.72	12.17	13.68
CL	12.02	14.16	3.33	**3.34**	3.49	2.44	11.72	13.52	14.49
DisPos	12.8	15.26	3.25	3.17	3.9	3.09	10.22	8.56	12.51
DT	11.62	13.38	3.49	3.37	3.96	3.24	11.97	13.38	14.43
TLP	11.96	13.98	3.33	2.98	3.65	2.98	12.02	13.5	13.83
ZS+OT	12.83	14.54	**3.5**	3.11	3.94	**3.26**	11.92	14.44	15.1
ZS+AT	11.99	14.11	3.41	3.03	3.46	2.54	11.9	14.11	14.2
Ours	**12.85**	15.21	3.24	3.18	**3.95**	3.04	**12.81**	**14.96**	15.29

Table 5: The results of each zero-shot translation direction on the PC-6 corpus. The notations denote the same meaning as in Table 2.

OPUS-6	Size
En-Kk	0.12M
En-Tr	0.39M
En-Ro	0.77M
En-Cs	0.82M
En-Ru	1.84M

Table 6: The statistics about the PC-6 corpus.

	γ_1	γ_2
IWSLT	0.4	0.001
IWSLT-b	0.2	0.002
PC-6	0.2	0.003
OPUS-7	0.3	0.01

Table 7: The hyperparameters γ_1 and γ_2 on each dataset.

α	β	zero Avg.
1	1	17.23
6	2	17.44
6	3	17.59
6	4	17.5

Table 8: The averaged BLEU with different α and β for the 'ZS+AT' system.

	kwps	ratio
ZS	199	1
SRA	118	0.59
SF	61	0.31
CL	94	0.47
ZS+OT	125	0.63
ZS+AT	61	0.31
Ours	58	0.29

Table 9: The training speed on the IWSLT dataset.