Addressing Asymmetry in Multilingual Neural Machine Translation with Fuzzy Task Clustering

Qian Wang[1,2] and **Jiajun Zhang**[1,2,3*]
[1]National Laboratory of Pattern Recognition, Institute of Automation, CAS, Beijing, China
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences , Beijing, China
[3]Beijing Academy of Artificial Intelligence, Beijing, China
{qian.wang, jjzhang}@nlpr.ia.ac.cn

Abstract

Multilingual neural machine translation (NMT) enables positive knowledge transfer among multiple translation tasks with a shared underlying model, but a unified multilingual model usually suffers from capacity bottleneck when tens or hundreds of languages are involved. A possible solution is to cluster languages and train individual model for each cluster. However, the existing clustering methods based on language similarity cannot handle the asymmetric problem in multilingual NMT, *i.e.*, one translation task A can benefit from another translation task B but task B will be harmed by task A. To address this problem, we propose a fuzzy task clustering method for multilingual NMT. Specifically, we employ task affinity, defined as the loss change of one translation task caused by the training of another, as the clustering criterion. Next, we cluster the translation tasks based on the task affinity, such that tasks from the same cluster can benefit each other. For each cluster, we further find out a set of auxiliary translation tasks that benefit the tasks in this cluster. In this way, the model for each cluster is trained not only on the tasks in the cluster but also on the auxiliary tasks. During training, we design a dynamic task sampling strategy that eliminate the negative influence of auxiliary tasks while exploit the positive knowledge of them. We conduct extensive experiments for one-to-many, many-to-one, and many-to-many translation scenarios to verify the effectiveness of our method.

1 Introduction

Neural machine translation (NMT) has achieved great success in recent years (Sutskever et al., 2014; Bahdanau et al., 2015; Vaswani et al., 2017). The conventional bilingual translation model well handles the translation task for a single language pair, but it is infeasible to train an individual model for each language pair since there are thousands of languages in the world. To improve the computational efficiency, researchers propose multilingual NMT that enables one model to handle multiple translation tasks (Ha et al., 2016; Johnson et al., 2017). Apart from the benefits for training and deployment, the shared underlying neural network in multilingual NMT also brings knowledge transfer among similar languages (Tan et al., 2019).

	A		B*		C*	
	en-it	en-nl	en-de	en-fr	en-zh	en-ja
Indiv	32.6	31.6	**28.1**	**38.8**	22.2	**13.6**
Joint	**34.3**	**34.3**	25.9	37.9	**25.3**	10.4

Table 1: Three kinds of task relationship. (A): Tasks benefit from each other when jointly trained. (B): Tasks are negatively influenced by each other. (C): The task en-zh benefits from en-ja but en-zh is harmful to en-ja. The models are trained on the IWSLT'17 dataset and results with * are reported in Xu and Yvon (2021).

Despite its simplicity, multilingual NMT model often suffers from representation bottleneck caused by language interference (Wang et al., 2020b; Wang and Zhang, 2022) when massive number of languages involved (Aharoni et al., 2019), which leads to performance degradation compared to bilingual translation models. Several methods have been proposed to break the bottleneck, such as designing language specific modules (Wang et al., 2019; Sachan and Neubig, 2018), hidden units (Wang et al., 2018) or routing path (Zhang et al., 2021). Among them, the most intuitive approach is grouping the languages into several clusters and training one multilingual NMT model for each cluster. These methods use similarity between language representations as clustering criterion, in which the representations come from language embedding vectors of a pretrained universal multilingual model or sparse language vectors of a multilingual knowledge base (Tan et al., 2019; Oncevay et al., 2020).

[*]Corresponding Author: Jiajun Zhang.

However, language similarity is not enough for modeling the relationship among translation tasks because of two reasons. First, a translation task includes a source language and a target language, while the language similarity only consider the feature of a single language. Second, the similarity metric cannot handle the asymmetric relationship between translation tasks. Zamir et al. (2018); Wang et al. (2022) show the asymmetry in multi-task learning and multilingual translation in which one task may benefit from another but not vice versa. Table 1 shows the three kinds of task relationships in multilingual NMT. Case (A) and (B) represent **symmetric** relationship in which the influence of the two tasks on each other are both positive or both negative. While in case (C), the relationship is **asymmetric** in which the task en-zh can benefit from the task en-ja but en-ja is harmed by en-zh when the two tasks are jointly trained.

In this work, we propose a fuzzy task clustering method for multilingual NMT to address the asymmetric problem among translation tasks. We use the task affinity (Fifty et al., 2021) as the clustering criterion, which measures the loss change of one task caused by the training of another task. We first train a universal multilingual model to obtain the affinity between each two tasks. Then we cluster tasks based on the affinity score such that tasks within each cluster can symmetrically benefit each other. For each cluster, we also select a set of auxiliary tasks that asymmetrically benefit the tasks in this cluster. Finally, we build separate models for each cluster, and train the model on both the tasks in the cluster and the corresponding auxiliary tasks. In this way, the model for each cluster can exploit the auxiliary tasks to facilitate training and improve the model quality. Compared to previous language clustering based methods, the proposed method improves positive knowledge transfer between tasks by introducing auxiliary tasks. To further address the increased burden of model training due to auxiliary tasks involved, we therefore propose a dynamic data sampling strategy, in which the sampling weights of auxiliary tasks gradually decrease to allow the optimization process to focus more on the tasks that used in inference.

To summarize, our method has the following advantages:

- The clustering criterion in our method directly measures the effect on one translation task when training another, which can better model the relationship between tasks compared to language similarity.
- The fuzzy clustering paradigm introduces auxiliary tasks for each cluster, breaking the barrier between clusters and improving positive knowledge transfer.

2 Problem Formulation

2.1 General Formulation

Given a set of translation tasks $\mathcal{T} = \{\tau_1, \dots, \tau_N\}$, we group the tasks into $K \in \{k \in \mathbb{N}^+ | k \leq N\}$ clusters $\mathcal{C}$:

$$\mathcal{C} = \left\{ c_1, \dots, c_K \;\middle|\; \begin{array}{l} c_k \neq \emptyset, \forall k \\ \bigcup_{k=1}^{K} c_k = \mathcal{T} \\ c_i \cap c_j = \emptyset, \forall i \neq j \end{array} \right\} \tag{1}$$

For each cluster c_k, we find out a set of auxiliary tasks from other clusters that benefit the tasks in c_k. By combining the auxiliary tasks and tasks in c_k, we obtain another cluster g_k. The set of g_k is denoted as:

$$\mathcal{G} = \left\{ g_1, \dots, g_K \;\middle|\; \begin{array}{l} \bigcup_{k=1}^{K} g_k = \mathcal{T} \\ c_k \subseteq g_k, \forall k \end{array} \right\} \tag{2}$$

where the tasks $\tau \in g_k \setminus c_k$ are the auxiliary tasks of c_k.

We then build a multilingual NMT model M_k for each cluster c_k to handle the translation tasks in c_k. The model M_k is trained on g_k, which consists of both the tasks from c_k and the corresponding auxiliary tasks, by minimizing the cross-entropy loss:

$$\arg\min_{M_k} \sum_{\tau \in g_k} \sum_{(\mathbf{x},\mathbf{y}) \in \tau} -\log p(\mathbf{y}|\mathbf{x}, M_k) \tag{3}$$

where $(\mathbf{x}, \mathbf{y})$ is a sentence pair from task τ.

For a predefined quality measure $Q(\tau|M)$ of a model M on task τ, our goal is to find the optimal clusters $\mathcal{C}$ and $\mathcal{G}$ that maximize the inference performance of all tasks:

$$\arg\max_{\mathcal{C},\mathcal{G},K} \sum_{k=1}^{K} \sum_{\tau \in c_k} Q(\tau|M_k) \tag{4}$$

2.2 Specialized Cases

We can see in this section that bilingual, multilingual, and clustering based methods are special cases of our formulation.

Bilingual Translation When $K = N$, namely we assign only one translation task for each cluster and train individual model for each task, the problem degrades into bilingual translation. The bilingual model for each cluster is trained with Equation (3).

Multilingual Translation When $K = 1$, a universal multilingual NMT model is trained for all tasks as in Johnson et al. (2017). To distinguish different languages, a special token called language tag is appended at the beginning of each source sentence.

Language Clustering for Multilingual NMT When $1 < K < N, \mathcal{C} = \mathcal{G}$, the tasks are grouped into separate clusters without auxiliary tasks. For example, Tan et al. (2019) first train a universal multilingual NMT model to obtain language embedding vectors. Based on the the embedding vectors, they cluster the languages into different groups and train individual model for each cluster.

3 Method

In this section, we first describe the translation task affinity used for modeling the asymmetric relationship between tasks (Section 3.1), followed by the translation task clustering method that clusters the tasks and finds out the auxiliary tasks for each cluster (Section 3.2). We finally build a model for each cluster and train the model with dynamic data sampling strategy (Section 3.3).

3.1 Translation Task Affinity

We first focus on the quality measure $Q(\tau|M)$ of a model M on task τ. A popular choice for modeling and optimizing the quality of a translation model is cross-entropy loss (Bahdanau et al., 2015; Vaswani et al., 2017). A lower cross-entropy indicates a better model. Therefore, we use the negative cross-entropy, namely the log-probability, to measure the quality of M on task τ:

$$Q(\tau|M) = \sum_{(\mathbf{x},\mathbf{y})\in\tau} \log p(\mathbf{y}|\mathbf{x}, M) \tag{5}$$

Obviously, training one model M for each task τ is a possible solution for Equation (4), since the training objective in Equation (3) is identical to maximizing $Q(\tau|M)$.

Previous works find out that the performance on some tasks can be improved when jointly trained (Tan et al., 2019; Standley et al., 2020; Chiang et al., 2022). Therefore, we want to find out which task can benefit another under the quality measure $Q(\tau|M)$ in multilingual training. Formally, given two tasks τ_i and τ_j, the affinity $Z_{j\to i}$ from τ_j to τ_i is defined by:

$$Z_{j\to i} = Q(\tau_i|M_1) - Q(\tau_i|M_2) \tag{6}$$

where M_1 is trained on tasks τ_i and τ_j, while M_2 is trained on task τ_i. A positive value $Z_{j\to i} > 0$ indicates that the task τ_j brings positive knowledge transfer to task τ_i, while a negative value indicates that the task τ_j is harmful to task τ_i. Note that $Z_{j\to i} \neq Z_{i\to j}$, and thus we call it **asymmetric task affinity**.

However, exhaustively searching over all possible combinations of $|\mathcal{T}|$ tasks requires training $2^{|\mathcal{T}|} - 1$ models, which is unaffordable. We therefore turn to an efficient approximation that measures, at each training step, the change of one task's loss caused by the optimization of another task (Fifty et al., 2021). The affinity from task j to task i at step t is defined as[1]:

$$\hat{Z}^t_{j\to i} = \log p(\mathbf{y}|\mathbf{x}, M^{t+1}) - \log p(\mathbf{y}|\mathbf{x}, M^t) \tag{7}$$

where $(\mathbf{x}, \mathbf{y})$ is a sentence pair sampled from task τ_i, and M^{t+1} is obtained by one-step training on task τ_j of model M^t.

Similar to Tan et al. (2019), we first train a universal model for all the language pairs. During training, we calculate the affinity $Z^t_{j\to i}$ with randomly and uniformly sampled tasks τ_i and τ_j at t-th step[2]. The affinity is then accumulated across all steps:

$$Z_{j\to i} \approx \hat{Z}_{j\to i} = \sum_{t=1}^{T} \hat{Z}^t_{j\to i} \tag{8}$$

With the affinity matrix $Z \in R^{|\mathcal{T}|\times|\mathcal{T}|}$, our goal in Equation (4) becomes maximizing the overall inter-task affinity as proved in Section A:

$$\underset{\mathcal{C},\mathcal{G},K}{\arg\max} \sum_{k=1}^{K} \sum_{\tau_i\in c_k} \sum_{\tau_j\in g_k} Z_{j\to i} \tag{9}$$

[1] In Fifty et al. (2021), they consider the ratio of the loss change to eliminate the scale discrepancy among different task losses: $Z^t_{j\to i} = 1 - \frac{L_j(\mathbf{x},\mathbf{y},M^{t+1})}{L_j(\mathbf{x},\mathbf{y},M^t)}$. Since the task loss scales are similar in multilingual translation, we only consider the absolute loss change.

[2] In multilingual NMT, a single batch can contain sentence pairs from multiple tasks (Johnson et al., 2017), which is unsuitable for Equation (7). We therefore iteratively use batches contain multiple tasks to stabilize training, and batches contain only one task to calculate $Z^t_{j\to i}$.

3.2 Translation Task Clustering

We now describe how to solve Equation (9). Similar to *reduction from Set-Cover* that chooses a subset covering all the tasks to minimize the overall cost (Standley et al., 2020), this problem is NP-hard in general. We therefore propose a greedy approximation that iteratively constructs the clusters $\mathcal{C}$ and $\mathcal{G}$ to minimize the overall affinity.

As shown in Figure 1(b), we first initialize $\mathcal{C}$ with N clusters, where each cluster contains only one task. With the affinity matrix Z, we build the corresponding N clusters of $\mathcal{G}$:

$$g_k = c_k \cup \{\tau_j | Z_{j\to i} > 0\} \quad \forall k = 1, \dots, N \tag{10}$$

where $\tau_i \in c_k$.

We define the score function $S(c_k, g_k)$ to measure the cluster affinity of c_k and corresponding g_k:

$$S(c_k, g_k) = \frac{1}{|g_k|} \sum_{\tau_i \in c_k} \sum_{\tau_j \in g_k} Z_{j\to i} \tag{11}$$

where $\frac{1}{|g_k|}$ is the normalization term for the number of tasks. We also define the rule of the union of two clusters c_1 and c_2 in $\mathcal{C}$, and g_1 and g_2 in $\mathcal{G}$:

$$\begin{aligned} c_1 \underset{\mathcal{C}}{\cup} c_2 =& \{\tau | \tau \in c_1 \text{ or } \tau \in c_2\} \\ g_1 \underset{\mathcal{G}}{\cup} g_2 =& (c_1 \underset{\mathcal{C}}{\cup} c_2) \cup \\ & \left\{ \tau_j \middle| \begin{array}{l} \tau_j \in g_1 \text{ or } \tau_j \in g_2 \\ Z_{j\to i} > 0, \forall \tau_i \in c_1 \underset{\mathcal{C}}{\cup} c_2 \end{array} \right\} \end{aligned} \tag{12}$$

As shown in Figure 1(c-e), for each two clusters, we form a union of them based on the above union rule and calculate the cluster affinity. The clusters with maximum overall affinity are selected. The clustering step is repeated until no union of two clusters can further improve the overall affinity, *i.e.*, for any of two clusters c_1 and c_2, we have:

$$S(c_1 \underset{\mathcal{C}}{\cup} c_2, g_1 \underset{\mathcal{G}}{\cup} g_2) \leq S(c_1, g_1) + S(c_2, g_2) \tag{13}$$

After iteration[3], the Equation (4) is solved and we obtain K clusters in $\mathcal{C}$ as well as K corresponding clusters in $\mathcal{G}$. We train one multilingual NMT model (Johnson et al., 2017) for the tasks in each cluster, where each model M_k is trained with tasks in g_k and is used for inference of tasks in c_k.

(a) Affinity Matrix Z

	τ_1	τ_2	τ_3
τ_1	1.1	0.2	0.2
τ_2	0.3	0.9	-0.2
τ_3	-0.2	0.1	1.2

(b) Initialize $\mathcal{C}$ and $\mathcal{G}$

k	$\mathcal{C}$	$\mathcal{G}$	$S(c,g)$
1	τ_1	τ_1, τ_2	0.7
2	τ_2	τ_1, τ_2, τ_3	0.4
3	τ_3	τ_1, τ_3	0.7

(c) Union clusters 1 and 2

k	$\mathcal{C}$	$\mathcal{G}$	$S(c,g)$
1	τ_1, τ_2	τ_1, τ_2	1.25
2	τ_3	τ_1, τ_3	0.7

(d) Union clusters 1 and 3

k	$\mathcal{C}$	$\mathcal{G}$	$S(c,g)$
1	τ_1, τ_3	τ_1, τ_3	1.15
2	τ_2	τ_1, τ_2, τ_3	0.4

(e) Union clusters 2 and 3

k	$\mathcal{C}$	$\mathcal{G}$	$S(c,g)$
1	τ_1	τ_1, τ_2	0.7
2	τ_2, τ_3	τ_1, τ_2, τ_3	0.8

Figure 1: An example of clustering 3 tasks. Based on the affinity matrix (a), we first initialize the clusters $\mathcal{C}$ and $\mathcal{G}$ (b), and then make union of each of two clusters (c-e). In this example, (c) brings highest overall affinity (1.95) and no union of two clusters can further improve the affinity (1.2 if the three tasks belong to one cluster). Thus the clusters in (c) is the clustering result.

3.3 Training with Dynamic Data Sampling

We now focus on the training of model M_k. The training objective, as described in Equation (3), is to minimize the overall cross-entropy loss for tasks in g_k. Since the training tasks in g_k contain both the inference tasks in c_k and other auxiliary tasks $g_k \setminus c_k$, the objective brings positive knowledge transfer and can improve the translation quality compared to that only trained with tasks in c_k. However, it also introduces extra burden for the model due to more auxiliary tasks involved.

We therefore propose a **dynamic data sampling** strategy that gradually decreases the weights of auxiliary tasks and focuses more on inference tasks. We start from the temperature based sampling (Conneau et al., 2020) that samples training data from each task $\tau \in g_k$ according to the data size of each task $|D_\tau|$ and a temperature term ρ:

$$\hat{P}_k(\tau) = \frac{q_\tau^{1/\rho}}{\sum\limits_{\pi \in g_k} q_\pi^{1/\rho}} \text{ where } q_\tau = \frac{|D_\tau|}{\sum\limits_{\pi \in g_k} |D_\pi|} \tag{14}$$

We re-scale the sampling distribution $P(\tau)$ for the auxiliary tasks $\tau \in g_k \setminus c_k$ as a function of the training epoch E:

$$P_k(\tau) = \begin{cases} \frac{1}{1+\lambda E} \frac{|c_k|}{|g_k|} \hat{P}_k(\tau) & \text{if } \tau \notin c_k \\ \hat{P}_k(\tau) & \text{if } \tau \in c_k \end{cases} \tag{15}$$

where λ is the decay rate of the sampling weight. The term $\frac{|c_k|}{|g_k|}$ considers the number of auxiliary tasks. If there are too many auxiliary tasks for

[3] In most of our experiments, the iteration can stop with a proper number of resulting K clusters. However, there is no theoretical guarantee for convergence and it may degrade into bilingual ($K = N$) or universal ($K = 1$) models. In practise, early-stop can be adopted when K satisfies a predefined constraints (Standley et al., 2020).

training the model, namely $|g_k| \gg |c_k|$, the weight of each auxiliary task should be smaller. Other dynamic data sampling and task weighting methods can also be used for prioritizing the inference tasks in c_k (Lin et al., 2019; Wang et al., 2020a; Mahapatra and Rajan, 2020).

4 Experiment Setup

4.1 Dataset

We evaluate our method on the TED 2020 Parallel Sentences Corpus collected by Reimers and Gurevych (2020). The dataset contains sentences from TED talks with their translations in more than 100 languages provided by a global community of volunteers.

For one-to-many (O2M) translation and many-to-one (M2O) translation scenarios, we select 28 languages (es, fr, ar, zh, ko, ru, tr, it, ja, he, pt, ro, vi, nl, hu, fa, pl, de, el, sr, bg, uk, hr, cs, id, th, sv, sk) $\leftrightarrow$ English that contain more than 100K sentence pairs[4] and the data statistics are shown in Supplementary Materials (Section B). We randomly select $4,000$ sentence pairs as validation set and $4,000$ sentence pairs as test set for each language pair. We use byte-pair encoding (BPE) (Sennrich et al., 2016) to encode all sentences and learn the BPE operation using sentencepiece (Kudo and Richardson, 2018), which results in a shared subword vocabulary containing 32K sub-word symbols.

For many-to-many (M2M) translation, we select 10 languages (es, fr, ar, ko, ru, tr, it, ja, he, pt) from the TED 2020 Parallel Sentences Corpus. Each two languages contains about 300K parallel sentences, which results in 90 translation directions. We preprocess the data in the same way as in O2M and M2O setting.

The multilingual datasets are sampled with temperature based strategy with $\rho = 5$, and the sampling weight decay rate in our method is set to $\lambda = 0.5$.

4.2 Model Settings

We conduct the experiments using the Transformer model (Vaswani et al., 2017) and implement our method based on the `fairseq` codebase[5]. For the O2M and M2O experiments, we use a 4-layer model[6] with embedding size 512 and FFN layer dimension 1024. For the M2M experiment, we adopt the `transformer_iwslt_de_en` configuration which represents 6-layer model. Each mini-batch contains roughly $16,384$ tokens. All models are trained with Adam optimizer (Kingma and Ba, 2015) on Nvidia 3090 GPUs. We use 2 GPUs for O2M and M2O translation, and 4 GPUs for M2M translation. We use SacreBLEU to measure the translation quality (Papineni et al., 2002; Post, 2018) and test the statistical significance by bootstrap resampling (Koehn, 2004).

4.3 Systems

The following methods are used in our experiments.

Bilingual (Bi.) We train a Transformer model for each task, which results in N individual models for N translation directions (Vaswani et al., 2017) .

Multilingual (Multi.) We train a universal multilingual Transformer model for all translation directions (Johnson et al., 2017).

Language Embedding Clustering (LEC) We cluster the languages based on the language embedding and train one multilingual model for each cluster. The number of clusters K is obtained with elbow method following Tan et al. (2019).

Hard Task Affinity Clustering (HTAC) We use the affinity instead of language embedding for clustering and no auxiliary tasks are used for training, namely we set $\lambda = +\infty$ in Equation (15), which is equivalent to $\mathcal{G} = \mathcal{C}$.

Fuzzy Task Affinity Clustering (FTAC) We use the affinity for clustering and incorporate auxiliary tasks with dynamic data sampling during training.

5 Results and Analyses

5.1 Clustering Results

For the **LEC** method (Tan et al., 2019), we obtain 9 clusters in O2M translation: {{sk, cs, hr, sr, pl}, {sv, uk}, {th, el, hu, vi, he, ar}, {id}, {bg, ro}, {de, fa, nl, pt, it, tr, fr, es}, {ru}, {zh}, {ja, ko}}. In M2O translation, there are 11 clusters: {{sk, cs}, {sv}, {th}, {id, vi}, {hr, bg, sr}, {uk, pl, ru}, {el}, {de, hu, nl, tr}, {fa, he, ar}, {ro, pt, it, fr, es}, {ja, ko, zh}}.

[4] We remove Brazilian Portuguese (pt-br) and Traditional Chinese (zh-tw) since they are similar to Portuguese (pt) and Simplified Chinese (zh-cn). we use zh to indicate zh-cn for simplicity.

[5] https://github.com/pytorch/fairseq

[6] We use 4-layer model because some bilingual models cannot converge with 6-layer setting.

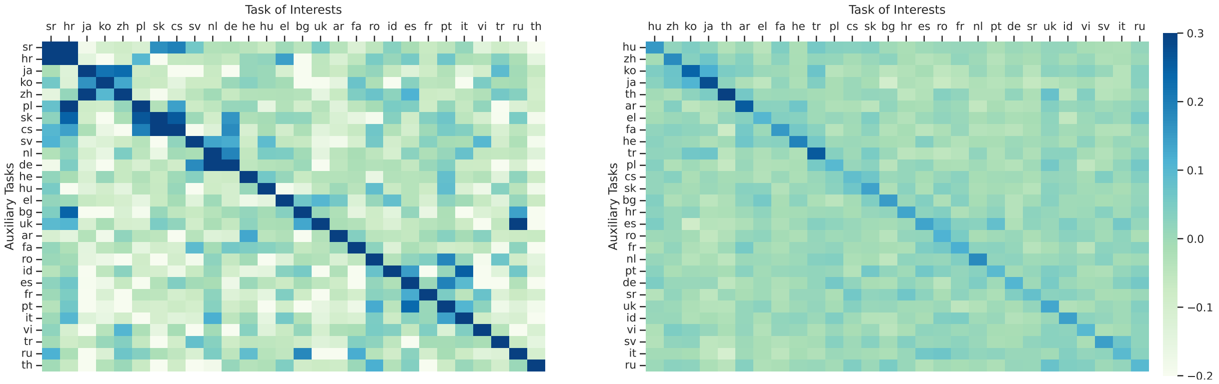

Figure 2: The task affinity between English to 28 languages (O2M, Left) and 28 languages to English (M2O, Right). Each cell (i, j) represents the affinity $Z_{i \to j}$ of task τ_i to task τ_j.

k	c_k (en→X)	$g_k \setminus c_k$ (en→X)
1	sr, hr	fr, uk, bg, sv, el, pl, cs, es, it, sk
2	ja, zh, ko	tr, vi, th
3	pl, sk, cs	hu, ru, he, fr
4	sv, nl, de	zh, id, sk, hr
5	he, hu	sv, ja, sk, de, ar
6	el, bg, uk	sk, es, fa, sr, th
7	ro, id, es, fr, pt, it	sv, cs, de, hr, sr, pl
8	ar, fa	hu, vi, de, nl
9	tr, vi	ja, zh, ar, th, ko
10	th, ru	ja, zh, el, ar, he, ko

Table 2: The clustering results of our method for one-to-many translation. Each language X denotes a translation task en→X.

k	c_k (X→en)	$g_k \setminus c_k$ (X→en)
1	hu	sk, ar, ro, pt, pl, hr, sv, ru, th, uk, fa, id, zh, it, fr, de, ja, ko, cs, he
2	zh, ko, ja, th	ar, uk, de, hu
3	ar, el, fa, he	ru, cs, pl, ro, es, hr, uk, sr, ja, id, bg
4	tr, pl, cs, sk	ru, sr, ko, ja
5	bg, hr	sk, ru, cs, pl, fa, nl, el, id
6	es, ro, fr, nl, pt	sk, id, bg
7	de, sr, uk	sk, sv, ru, cs, pl, ro, ar, hr, fa, nl, ko, it, zh, el, id, he
8	id, vi	th, fr, pl, hr, uk, de, fa, sr, ko, it, ja, zh, tr, he
9	sv, it, ru	sk, fr, pl, uk, nl, ja, pt, he

Table 3: The clustering results of our method for many-to-one translation. Each language X denotes a translation task X→-en.

For our task affinity based method (**FTAC**), we first show the task affinity in one-to-many and many-to-one scenarios in Figure 2. Obviously, the left part (O2M) and the right part (M2O) share one common pattern that the cells on the diagonal line are darker, which indicates that each task contributes more on itself. Besides, there are more differences between O2M and M2O models which leads to several interesting findings:

- We find that the cells on the diagonal line in O2M are darker than M2O models, and the cells beyond diagonal line are lighter in O2M model. The observation indicates that a task in O2M model relies more on itself and the positive knowledge transfer is more common in M2O multilingual model. This may be because the burden of generation is mainly on the decoder (Dabre et al., 2020), and the tasks in a M2O model share one target language. Thus the tasks in M2O translation can benefit each other and reduce the generation burden.
- The affinity between tasks better correlates with linguistic similarity in O2M model. For example, the Serbian (sr) and Croatian (hr) are similar languages and the affinity between en→sr and en→hr is quite high. Similar to language embedding (Tan et al., 2019), the affinity also captures the regional, cultural, and historical influences in O2M model (see the affinities between ja, ko, and zh).
- The affinities in M2O model cannot reflect linguistic proximities. As claimed above, the positive knowledge transfer mainly occurs in the decoder to facilitate generation. The differences in the target side between M2O translation tasks are imperceptible and affected by factors like data sizes, training schedule or optimization.

Based on the affinity matrix, we cluster the languages using the method described in Section 3.2 and show the clustering results for O2M multilingual translation in Table 2 and M2O translation in

	One-to-Many							Many-to-One						
	Bi.	**Multi.**	**LEC**	**HTAC**	**FTAC**	Δ_1	Δ_2	**Bi.**	**Multi.**	**LEC**	**HTAC**	**FTAC**	Δ_1	Δ_2
en↔es	40.5	38.9	41.3	42.0	**42.5**	+1.2	+3.6†	43.6	46.1	46.2	46.4	**48.4**	+2.2	+2.3†
en↔fr	41.4	39.6	42.2	42.8	**43.3**	+1.1	+3.7†	37.9	40.3	40.0	40.5	**42.1**	+2.1	+1.8†
en↔ar	16.3	14.5	17.2	17.2	**17.7**	+0.5	+3.2†	32.9	34.6	36.3	36.5	**37.9**	+1.6	+3.3†
en↔zh	16.6	14.6	16.6	16.5	**17.0**	+0.4	+2.4†	22.5	25.4	25.2	25.8	**26.6**	+1.4	+1.2†
en↔ko	6.5	6.4	7.6	7.4	**7.9**	+0.3	+1.5†	19.7	21.7	22.6	23.2	**23.7**	+1.1	+2.0†
en↔ru	20.9	18.4	20.9	20.9	**21.2**	+0.3	+2.8†	25.4	28.8	28.4	28.8	**28.9**	+0.5	+0.1
en↔tr	17.0	15.9	17.5	18.6	**19.1**	+1.6	+3.2†	27.1	28.0	28.7	29.5	**30.5**	+1.8	+2.5†
en↔it	33.4	32.7	35.0	35.8	**35.9**	+0.9	+3.2†	37.5	40.8	40.8	41.0	**41.3**	+0.5	+0.5‡
en↔ja	14.5	14.0	17.0	17.0	**17.5**	+0.5	+3.5†	15.3	17.4	18.3	19.0	**19.6**	+1.3	+2.2†
en↔he	26.9	24.4	28.1	28.6	**29.1**	+1.0	+4.7†	38.2	38.9	40.5	40.7	**42.9**	+2.4	+4.0†
en↔pt	37.6	35.9	38.5	39.4	**39.9**	+1.4	+4.0†	41.1	43.5	44.0	44.3	**45.9**	+1.9	+2.4†
en↔ro	27.7	26.6	28.5	29.5	**30.0**	+1.5	+3.4†	37.2	40.1	39.6	39.9	**41.9**	+2.3	+1.8†
en↔vi	29.0	27.5	30.7	30.8	**31.2**	+0.5	+3.7†	29.6	32.0	32.2	32.2	**32.7**	+0.5	+0.7†
en↔nl	31.7	29.8	32.7	32.9	**33.4**	+0.7	+3.6†	36.8	38.5	38.6	38.8	**40.1**	+1.5	+1.6†
en↔hu	17.8	14.8	17.0	17.8	**18.3**	+1.3	+3.5†	27.1	27.3	27.6	27.1	**28.7**	+1.1	+1.4†
en↔fa	18.2	17.7	19.5	19.6	**20.1**	+0.6	+2.4†	29.8	31.9	32.1	33.0	**34.5**	+2.4	+2.6†
en↔pl	15.6	15.3	18.2	18.4	**18.9**	+0.7	+3.6†	23.2	27.0	26.8	27.3	**28.1**	+1.3	+1.1†
en↔de	26.6	24.9	27.3	28.3	**28.8**	+1.5	+3.9†	33.8	35.7	36.1	35.8	**37.4**	+1.3	+1.7†
en↔el	32.0	28.8	33.9	33.7	**34.2**	+0.3	+5.4†	39.6	41.0	39.6	41.5	**42.5**	+2.9	+1.5†
en↔sr	22.0	21.9	25.5	24.3	**24.8**	-0.7	+2.9†	37.6	40.2	41.9	**42.0**	41.7	-0.2	+1.5†
en↔bg	32.6	30.8	34.6	34.6	**35.1**	+0.5	+4.3†	39.2	42.5	42.7	**42.8**	42.7	+0.0	+0.2
en↔uk	18.9	18.3	21.5	21.1	**21.6**	+0.1	+3.3†	27.4	31.2	30.9	31.4	**31.5**	+0.6	+0.3
en↔hr	27.4	26.0	**30.6**	29.5	30.3	-0.3	+4.3†	38.0	42.7	**44.1**	43.6	42.8	-1.3	+0.1
en↔cs	20.6	19.0	23.1	23.3	**23.8**	+0.7	+4.8†	31.1	34.5	33.9	36.3	**36.7**	+2.8	+2.2†
en↔id	31.5	27.9	31.5	31.6	**32.6**	+1.1	+4.7†	31.7	33.3	34.5	34.6	**35.4**	+0.9	+2.1†
en↔th	17.8	16.3	18.5	19.3	**19.8**	+1.3	+3.5†	24.9	27.1	24.9	28.1	**29.1**	+4.2	+2.0†
en↔sv	34.9	30.7	33.6	36.5	**37.2**	+3.6	+6.5†	40.4	42.0	40.4	43.4	**44.4**	+4.0	+2.4†
en↔sk	20.6	19.5	25.2	24.3	**24.8**	-0.4	+5.3†	29.9	34.9	36.1	37.4	**38.1**	+2.0	+3.2†
Average	24.9	23.3	26.2	26.5	**27.0**	+0.8	+3.7	32.1	34.6	34.8	35.4	**36.3**	+1.6	+1.7

Table 4: BLEU scores of one-to-many (O2M, Left) and many-to-one (M2O, Right) translation with different methods. The column Δ_1 and Δ_2 are the improvements of **FTAC** compared to **LEC** and **Multi.** respectively. † and ‡ indicate the corresponding improvement is statistically significant with $p < 0.01$ and 0.05 respectively.

Table 3. We find that the clusters in M2O translation contain more auxiliary tasks, which proves that the M2O translation can benefit more from multilingual training.

For many-to-many translation, we show the affinity heatmap and clustering results in Supplementary Materials (Section C).

5.2 O2M and M2O Translation Quality

We present the translation quality of different methods in Table 4. By comparing different methods, several observations can be found.

- Our **FTAC** method can well address the asymmetry in multilingual NMT and outperforms the **Multi.** baseline in all translation tasks.
- Clustering methods perform better in O2M translation. The **LEC** method outperforms the multilingual baseline by 2.9 BLEU in O2M translation but only 0.2 BLEU in M2O translation. Similarly, the **HTAC** method achieves improvements of 3.2 BLEU and 0.8 BLEU in O2M and M2O respectively. The results prove that the burden of generation caused by number of target languages involved is important in O2M translation, and the problem can be well addressed by clustering the tasks into different groups.
- Clustering based on task affinity performs better than based on language embedding. Compared to **LEC** method, **HTAC** achieves improvements of 0.3 BLEU in O2M setting and 0.6 BLEU in M2O setting.
- The asymmetry feature of task affinity that leads to fuzzy clustering is important, and the auxiliary tasks in each cluster brings large improvements, especially for M2O setting. The **FTAC** method with auxiliary tasks outperforms the **HTAC** method without auxiliary tasks by 0.5 BLEU and 0.9 BLEU in O2M and M2O respectively.

	→es	→fr	→ar	→ko	→ru	→tr	→it	→ja	→he	→pt
es→	-	35.9/+3.9	12.2/+3.8	16.7/+3.9	16.9/+3.9	15.0/+4.4	30.3/+4.0	15.2/+5.3	18.8/+6.2	31.3/+6.3
fr→	28.9/+4.6	-	11.4/+2.6	16.6/+3.4	16.3/+4.5	14.0/+4.1	28.3/+4.1	14.7/+5.3	18.3/+5.3	28.8/+5.2
ar→	23.8/+2.4	27.8/+4.0	-	15.2/+2.4	14.0/+3.3	12.3/+3.2	21.8/+3.8	13.3/+2.3	15.5/+4.5	22.2/+5.4
ko→	15.5/+2.4	22.4/+2.3	6.8/+1.6	-	10.4/+2.0	9.8/+2.1	15.6/+2.5	14.4/+3.0	10.0/+2.2	14.8/+2.8
ru→	20.3/+3.5	27.1/+3.4	8.2/+1.9	14.9/+1.9	-	11.2/+2.3	20.0/+3.1	13.3/+3.8	13.2/+3.6	19.5/+4.9
tr→	21.0/+3.4	26.9/+3.0	9.4/+2.2	15.4/+2.8	12.6/+2.9	-	19.8/+3.6	14.0/+3.6	13.9/+3.0	20.4/+4.2
it→	30.2/+3.6	34.9/+3.3	11.3/+3.5	16.7/+3.8	16.4/+4.4	14.1/+2.3	-	14.8/+5.0	18.2/+5.1	29.1/+6.0
ja→	13.1/+1.5	19.8/+2.2	5.7/+1.3	13.4/+1.8	8.6/+0.3	8.6/+1.2	13.6/+1.8	-	7.8/+1.9	12.9/+2.5
he→	26.8/+4.8	31.9/+5.6	11.8/+3.2	16.4/+2.2	16.4/+3.8	13.5/+2.6	24.7/+4.3	14.5/+2.6	-	26.1/+6.1
pt→	32.5/+4.8	36.3/+3.7	11.9/+4.4	17.1/+4.6	16.4/+4.9	14.2/+5.3	30.2/+4.4	15.4/+5.3	18.7/+6.1	-

Table 5: Many-to-many (M2M) translation quality measured by BLEU score. We compare our method (**FTAC**) with the multilingual baseline (**Multi.**). The BLEU scores of **FTAC** and the improvements Δ are reported with the format of **FTAC**/Δ.

5.3 M2M Translation Quality

For many-to-many setting, the language embedding based clustering methods are not suitable since the source language and target language in a translation task may be clustered into different groups. On the other hand, our method well handles M2M multilingual translation by directly clustering different tasks based on the affinity between them.

We compare our method (**FTAC**) with the multilingual baseline (**Multi.**) and the results are shown in Table 5. Our method consistently outperforms the baseline method by up to $+6.3$ BLEU and $+3.6$ BLEU on average. The results in M2M translation show that our method can well handle M2M translation.

5.4 The Effects of Auxiliary Tasks

To further understand the performance gain contributed by auxiliary tasks, we analyze the effects of different sampling strategies for auxiliary tasks. Besides the **HTAC** and **FTAC** methods, we also compare the following strategies:

- **Rand.** We randomly select $|g_k \setminus c_k|$ auxiliary tasks for each cluster c_k.
- **Temp.** We use vanilla temperature based sampling for all tasks by setting $P_k(\tau) = \hat{P}_k(\tau)$ in Equation (15).
- **Fix.** We set $\lambda = 0$ in Equation (15), namely the sampling weights only correlate with the number of auxiliary tasks involved.

Method	**HTAC**	**Rand.**	**Temp.**	**Fix.**	**FTAC**
Result	35.4	33.9	35.0	35.1	36.3

Table 6: Results of different sampling strategies for auxiliary task data in one-to-many translation (measured by average BLEU).

Table 6 shows the average BLEU of different methods in one-to-many translation. The detailed results are shown in Supplementary Materials (Section D). By comparing **HTAC** and **Rand.**, we find that arbitrarily adding auxiliary tasks significantly hurts the performance by 1.9 BLEU, which proves the effectiveness of our **FTAC** method for selecting proper auxiliary tasks. Besides, we find that the **Temp.** and **Fix.** methods also perform worse than **HTAC**, which indicates that the auxiliary tasks bring extra burden for generation although they are correlated with the inference tasks. By gradually decreasing the weights of auxiliary tasks during training, the **FTAC** method can focus more on the inference tasks and bring better results.

6 Conclusion

In this work, we have proposed a fuzzy task clustering method to address the asymmetric problem in multilingual NMT based on task affinity. The task affinity is defined by the loss change of one task after a step training of another task. Based on the affinity, we cluster translation tasks, each of which contains tasks that are symmetric. Each cluster is further equipped with auxiliary tasks that can benefit the model training of this cluster. Experiments show that our fuzzy task clustering method significant outperforms the strong baselines. We also show the effectiveness of incorporating auxiliary tasks in a multilingual translation model. In the future, we plan to explore more efficient and effective clustering criterion by exploiting large-scale pre-trained multilingual models.

Acknowledgements

This research work is supported by National Key R&D Program of China (2020AAA0105200) and the Natural Science Foundation of China under Grant 62122088.

References

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual 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 3874–3884. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Ting-Rui Chiang, Yi-Pei Chen, Yi-Ting Yeh, and Graham Neubig. 2022. Breaking down multilingual machine translation. In *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 2766–2780. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 8440–8451. Association for Computational Linguistics.

Raj Dabre, Chenhui Chu, and Anoop Kunchukuttan. 2020. A survey of multilingual neural machine translation. *ACM Comput. Surv.*, 53(5):99:1–99:38.

Chris Fifty, Ehsan Amid, Zhe Zhao, Tianhe Yu, Rohan Anil, and Chelsea Finn. 2021. Efficiently identifying task groupings for multi-task learning. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 27503–27516.

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

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.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing , EMNLP 2004, A meeting of SIGDAT, a Special Interest Group of the ACL, held in conjunction with ACL 2004, 25-26 July 2004, Barcelona, Spain*, pages 388–395. ACL.

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, EMNLP 2018: System Demonstrations, Brussels, Belgium, October 31 - November 4, 2018*, pages 66–71. Association for Computational Linguistics.

Xingyu Lin, Harjatin Singh Baweja, George Kantor, and David Held. 2019. Adaptive auxiliary task weighting for reinforcement learning. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 4773–4784.

Debabrata Mahapatra and Vaibhav Rajan. 2020. Multi-task learning with user preferences: Gradient descent with controlled ascent in pareto optimization. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 6597–6607. PMLR.

Arturo Oncevay, Barry Haddow, and Alexandra Birch. 2020. Bridging linguistic typology and multilingual machine translation with multi-view language representations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 2391–2406. 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, July 6-12, 2002, Philadelphia, PA, USA*, pages 311–318. ACL.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers, WMT 2018, Belgium, Brussels, October 31 - November 1, 2018*, pages 186–191. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 4512–4525. Association for Computational Linguistics.

Devendra Singh Sachan and Graham Neubig. 2018. Parameter sharing methods for multilingual self-attentional translation models. In *Proceedings of the Third Conference on Machine Translation: Research Papers, WMT 2018, Belgium, Brussels, October 31 - November 1, 2018*, pages 261–271. Association for Computational Linguistics.

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*. The Association for Computer Linguistics.

Trevor Standley, Amir Roshan Zamir, Dawn Chen, Leonidas J. Guibas, Jitendra Malik, and Silvio Savarese. 2020. Which tasks should be learned together in multi-task learning? In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 9120–9132. PMLR.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112.

Xu Tan, Jiale Chen, Di He, Yingce Xia, Tao Qin, and Tie-Yan Liu. 2019. Multilingual neural machine translation with language clustering. 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 2019, Hong Kong, China, November 3-7, 2019*, pages 963–973. Association for Computational Linguistics.

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 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Qian Wang and Jiajun Zhang. 2022. Parameter differentiation based multilingual neural machine translation. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pages 11440–11448. AAAI Press.

Qian Wang, Jiajun Zhang, and Chengqing Zong. 2022. Synchronous inference for multilingual neural machine translation. *IEEE ACM Trans. Audio Speech Lang. Process.*, 30:1827–1839.

Xinyi Wang, Yulia Tsvetkov, and Graham Neubig. 2020a. Balancing training for multilingual neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 8526–8537. Association for Computational Linguistics.

Yining Wang, Jiajun Zhang, Feifei Zhai, Jingfang Xu, and Chengqing Zong. 2018. Three strategies to improve one-to-many multilingual translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2955–2960. Association for Computational Linguistics.

Yining Wang, Long Zhou, Jiajun Zhang, Feifei Zhai, Jingfang Xu, and Chengqing Zong. 2019. A compact and language-sensitive multilingual translation method. 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 1213–1223. Association for Computational Linguistics.

Zirui Wang, Zachary C. Lipton, and Yulia Tsvetkov. 2020b. On negative interference in multilingual models: Findings and A meta-learning treatment. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 4438–4450. Association for Computational Linguistics.

Jitao Xu and François Yvon. 2021. One source, two targets: Challenges and rewards of dual decoding. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 8533–8546. Association for Computational Linguistics.

Amir Roshan Zamir, Alexander Sax, William B. Shen, Leonidas J. Guibas, Jitendra Malik, and Silvio Savarese. 2018. Taskonomy: Disentangling task transfer learning. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 3712–3722. Computer Vision Foundation / IEEE Computer Society.

Biao Zhang, Ankur Bapna, Rico Sennrich, and Orhan Firat. 2021. Share or not? learning to schedule language-specific capacity for multilingual translation. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. 2020. Improving massively multilingual neural machine translation and zero-shot translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 1628–1639. Association for Computational Linguistics.

Language	es	fr	ar	zh	ko	ru	tr
Data Size	413	407	404	399	396	386	374
Language	it	ja	he	pt	ro	vi	nl
Data Size	370	363	349	326	325	323	317
Language	hu	fa	pl	de	el	sr	bg
Data Size	305	302	297	294	267	258	247
Language	uk	hr	cs	id	th	sv	sk
Data Size	206	196	169	163	159	120	105

Table 7: The data size (K) of 28 langusges X↔English used in M2O and O2M translation.

	es	fr	ar	ko	ru	tr	it	ja	he	pt
es	-	401	399	393	384	371	368	362	347	320
fr	401	-	396	390	382	369	365	360	344	317
ar	399	396	-	389	381	368	365	359	344	317
ko	393	390	389	-	378	363	362	356	342	314
ru	384	382	381	378	-	356	357	355	341	307
tr	371	369	368	363	356	-	340	338	324	300
it	368	365	365	362	357	340	-	340	330	296
ja	362	360	359	356	355	338	340	-	330	290
he	347	344	344	342	341	324	330	330	-	282
pt	320	317	317	314	307	300	296	290	282	-

Table 8: The data size (K) used in many-to-many (M2M) translation.

A The Clustering Objective

We want to maximize the overall performance of all tasks by finding the clusters $\mathcal{C}$ and $\mathcal{G}$:

$$\begin{aligned} &\arg\max_{\mathcal{C},\mathcal{G},K} \sum_{k=1}^{K} \sum_{\tau_i \in c_k} Q(\tau_i | M_k) \\ &s.t.\ \ M_k = \arg\min_{M_k} \sum_{\tau_j \in g_k} \sum_{(\mathbf{x},\mathbf{y}) \in \tau_j} -\log p(\mathbf{y}|\mathbf{x}, M_k) \end{aligned} \tag{16}$$

Assume that the quality $Q(\tau_i|M_k)$ on task τ_i with the model trained on tasks $\tau_j \in g_k$ is an approximation of the average of the model qualities on task τ_i with the models trained on task τ_j and task τ_i:

$$Q(\tau_i|M_k) \approx \frac{1}{|g_k|} \sum_{\tau_j \in g_k} Q(\tau_i | M_{\tau_i,\tau_j}) \tag{17}$$

	HTAC	Rand.	Temp.	Fix.	FTAC
es→en	46.4	45.1	46.8	47.4	**48.4**
fr→en	40.5	39.2	40.5	41.0	**42.1**
ar→en	36.5	34.4	36.8	36.4	**37.9**
zh→en	25.8	24.8	25.7	25.6	**26.6**
ko→en	23.2	22.0	22.8	22.8	**23.7**
ru→en	28.8	27.8	28.2	27.2	**28.9**
tr→en	29.5	28.3	28.9	29.8	**30.5**
it→en	41.0	39.6	40.4	39.5	**41.3**
ja→en	19.0	17.8	18.6	18.5	**19.6**
he→en	40.7	38.5	40.8	41.4	**42.9**
pt→en	44.3	42.8	44.4	44.9	**45.9**
ro→en	39.9	38.7	40.7	40.7	**41.9**
vi→en	32.2	32.1	31.0	31.4	**32.7**
nl→en	38.8	37.5	38.8	38.9	**40.1**
hu→en	27.1	27.5	25.8	27.6	**28.7**
fa→en	33.0	31.6	33.2	32.6	**34.5**
pl→en	27.3	25.9	26.8	27.5	**28.1**
de→en	35.8	37.0	36.9	36.3	**37.4**
el→en	41.5	39.1	40.8	40.9	**42.5**
sr→en	**42.0**	40.7	40.3	40.5	41.7
bg→en	**42.8**	40.3	42.5	41.2	42.7
uk→en	31.4	30.9	31.5	30.6	**31.5**
hr→en	**43.6**	39.5	43.3	41.2	42.8
cs→en	36.3	33.4	34.6	36.0	**36.7**
id→en	34.6	33.7	32.7	34.0	**35.4**
th→en	28.1	26.6	27.4	27.9	**29.1**
sv→en	43.4	41.6	42.9	42.3	**44.4**
sk→en	37.4	33.5	35.5	37.3	**38.1**
Average	35.4	33.9	35.0	35.1	**36.3**

Table 9: BLEU score of many-to-one (M2O) translation with different data sampling strategies.

where M_{τ_i,τ_j} is the model trained on tasks τ_j and τ_i. Then the objective becomes:

$$\begin{aligned} &\arg\max_{\mathcal{C},\mathcal{G},K} \sum_{k=1}^{K} \sum_{\tau_i \in c_k} Q(\tau_i | M_k) \\ \approx &\arg\max_{\mathcal{C},\mathcal{G},K} \sum_{k=1}^{K} \sum_{\tau_i \in c_k} \frac{1}{|g_k|} \sum_{\tau_j \in g_k} Q(\tau_i | M_{\tau_i,\tau_j}) \end{aligned} \tag{18}$$

Since the task $\tau_i \in c_k$ must appear in g_k, the term $Q(\tau_i|M_{\tau_i})$ is irrelevant to the $\arg\max$ operation:

$$\begin{aligned} &\arg\max_{\mathcal{C},\mathcal{G},K} \sum_{k=1}^{K} \sum_{\tau_i \in c_k} Q(\tau_i | M_k) \\ \approx &\arg\max_{\mathcal{C},\mathcal{G},K} \sum_{k=1}^{K} \sum_{\tau_i \in c_k} \sum_{\tau_j \in g_k} \left(Q(\tau_i | M_{\tau_i,\tau_j}) - Q(\tau_i | M_{\tau_i})\right) \\ = &\arg\max_{\mathcal{C},\mathcal{G},K} \sum_{k=1}^{K} \sum_{\tau_i \in c_k} \sum_{\tau_j \in g_k} Z_{j \to i} \end{aligned} \tag{19}$$

k	c_k	$g_k \setminus c_k$
1	ja-ru	it-ru, tr-ru, ja-it, it-pt, ar-ru, pt-ru, ko-ru, ja-fr, ru-fr, he-fr, fr-ru, es-ru, ar-es
2	ar-es	es-it, ja-es, ru-es, ar-it, pt-es, tr-es, es-ar, ar-pt, ko-es, fr-pt, ar-ko, fr-es
3	he-tr,ja-tr	fr-tr, ru-tr, es-tr, it-tr, pt-tr
4	pt-fr,tr-fr	ar-fr, it-fr, es-fr, ko-fr
5	ru-tr,it-tr	ar-tr, he-tr, ja-tr
6	he-ja,ar-ja	tr-ja, ko-ja, it-ja, pt-ja
7	ko-he,tr-he	ja-he, ru-he, es-he, ar-he, it-he
8	ru-ar,fr-ar	tr-ar, es-ar, ko-ar, it-ar
9	he-es,ja-es,it-es	ru-es, pt-es, ko-es, ar-es, fr-es
10	tr-ru,he-ru,es-ru	ar-ru, pt-ru, fr-ru
11	es-it,ru-it,ja-it	he-it, pt-it, ko-fr, fr-es, pt-es, he-it, fr-it, pt-it, tr-fr
12	pt-he,ru-he,ja-he,it-he	ko-he, tr-he, fr-ar
13	es-fr,he-fr,ru-fr	ar-fr, it-fr, pt-fr
14	ja-ar,tr-ar,ko-ar	ru-ar, fr-ar, es-ar, he-ar, it-ar
15	tr-pt,ja-pt,fr-pt,he-pt	ko-pt, it-pt, tr-it, ru-pt, es-pt
16	he-ko,ru-ko,ja-ko	tr-ja, ja-ar, pt-ko, es-tr, ar-ko, ru-ar, tr-ko
17	es-he,ar-he,fr-he	tr-he, it-he, es-ru, pt-he, ru-he, ko-he, ja-he
18	pt-es,ru-es,ko-es,tr-es,fr-es	ja-es, it-es, ko-fr, ru-ar
19	es-ar,he-ar,it-ar,pt-ar	ja-ar, ko-ja, tr-ar, fr-ar, pt-fr, it-pt, ko-ar
20	it-fr,ko-fr,ar-fr,ja-fr	es-fr, pt-fr, tr-fr, ru-fr
21	pt-ru,ko-ru,fr-ru,it-ru,ar-ru	es-ru, it-pt, he-ru
22	es-tr,pt-tr,ko-tr,fr-tr,ar-tr	ja-tr, ru-tr
23	it-ko,tr-ko,ar-ko,fr-ko,es-ko,pt-ko	ja-ko, he-ko, ru-ko
24	ru-pt,it-pt,es-pt,ar-pt,ko-pt	es-fr, ru-it, pt-it, ja-pt, fr-pt, he-pt
25	pt-ja,es-ja,tr-ja,ru-ja,fr-ja,ko-ja,it-ja	es-ko, ar-ja
26	he-it,ko-it,tr-it,fr-it,ar-it,pt-it	fr-es, ko-fr, ru-it, ru-pt, es-it

Table 10: The clustering results of our method for many-to-many translation.

B Data Statistics and Pre-processing

In our experiments, we use the data from the TED 2020 Parallel Sentences Corpus[7], which contains sentences from TED talks with their translations provided by a global community of volunteers[8].

For one-to-many and many-to-one translation, we select 28 languages and the data statistics are shown in Table 7. For many-to-many translation, we select 10 languages and the data statistics are shown in Table 8.

For the data pre-processing, we use Jieba[9] for segmenting Chinese sentences, Mecab[10] for Japanese and Korean, and Moses[11] for other languages.

C M2M Clustering

We show the task affinities of many-to-many (M2M) translation in Figure 3 and the clustering results in Table 10. From figure 3, we find that the tasks with the same target languages share higher affinities compared to the tasks with the same source languages. From the clustering results in Table 10, we also find that the tasks with the same target languages are more likely to be in the same cluster or serve as auxiliary tasks.

D The Effects of Auxiliary Tasks

The detailed BLEU scores of different data sampling strategies for auxiliary tasks are shown in Table 9.

E Results on OPUS dataset

In our experiments, we randomly split the data into training/validation/test set. However, there can be overlap between sentences in train and test/validation sets of different language pairs. For example, one English sentence X in the test set of task en-de may occur in the training set of en-fr. To make the results more convincing, we evaluate the

[7] https://github.com/UKPLab/sentence-transformers/blob/master/docs/datasets/TED2020.md.

[8] https://www.ted.com/participate/translate

[9] https://github.com/fxsjy/jieba

[10] https://taku910.github.io/mecab/

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

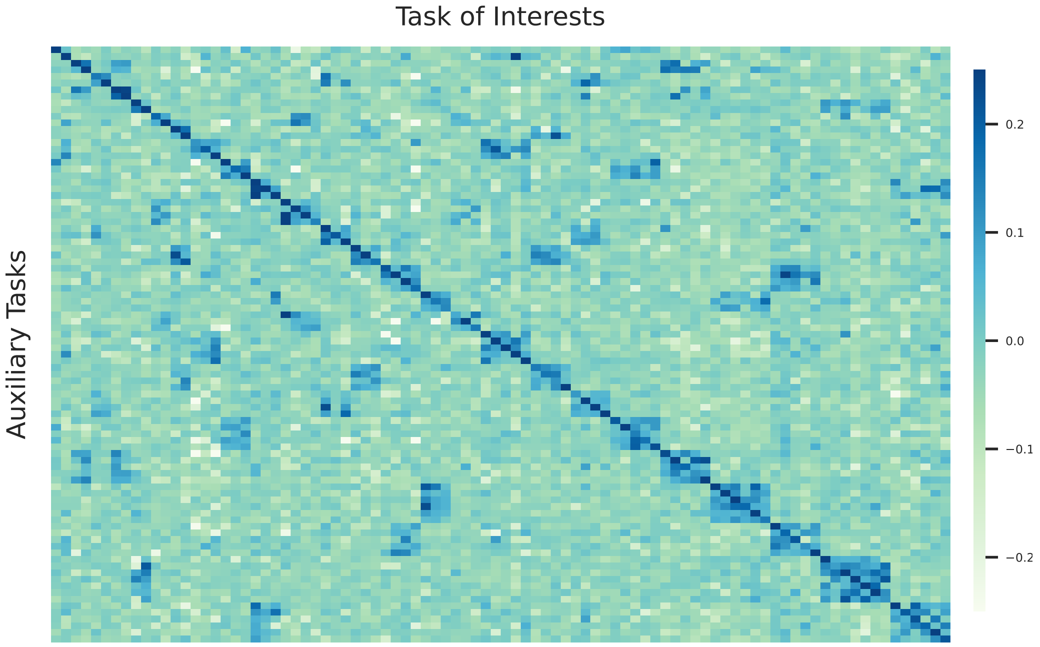

Figure 3: The affinity heatmap between tasks in many-to-many translation. The orders of tasks in X and Y axes are identical to the tasks of c_k in Table 10.

	Multi.	**LEC**	**FTAC**	Δ_1	Δ_2
es→en	25.1	27.0	28.9	+1.9	+3.8
fr→en	16.1	18.4	20.0	+1.6	+3.9
ar→en	19.2	21.3	23.3	+2.0	+4.1
zh→en	10.0	10.9	11.5	+0.6	+1.5
ko→en	12.6	14.5	15.1	+0.6	+2.5
ru→en	15.4	16.4	17.7	+1.3	+2.3
tr→en	21.1	22.2	24.3	+2.1	+3.2
it→en	20.6	23.1	23.8	+0.7	+3.2
ja→en	11.9	12.9	14.1	+1.2	+2.2
he→en	28.9	30.9	33.7	+2.8	+4.8
pt→en	24.8	27.0	28.3	+1.3	+3.5
ro→en	27.3	28.6	30.6	+2.0	+3.3
vi→en	19.9	22.8	23.2	+0.4	+3.3
nl→en	21.3	23.2	24.6	+1.4	+3.3
hu→en	19.3	20.1	22.2	+2.1	+2.9
fa→en	13.1	15.4	17.0	+1.6	+3.9
pl→en	20.0	20.7	22.7	+2.0	+2.7
de→en	13.7	16.8	17.3	+0.5	+3.6
el→en	22.2	25.3	24.1	-1.2	+1.9
sr→en	24.0	26.9	25.7	-1.2	+1.7
bg→en	23.8	25.6	26.1	+0.5	+2.3
uk→en	21.4	22.3	22.4	+0.1	+1.0
hr→en	23.6	27.0	26.2	-0.8	+2.6
cs→en	20.7	22.5	23.3	+0.8	+2.6
id→en	24.9	29.5	28.8	-0.7	+3.9
th→en	16.9	18.0	18.9	+0.9	+2.0
sv→en	17.7	19.6	20.8	+1.2	+3.1
sk→en	20.5	22.6	22.7	+0.1	+2.2
Average	19.9	21.8	22.8	+0.9	+2.9

Table 11: BLEU score of many-to-one (M2O) translation on the OPUS test set.

M2O model on the OPUS-100 (Zhang et al., 2020) multilingual test set which contains no overlap sentences in different tasks, and the results are shown in Table 11.