Unsupervised Neural Machine Translation for Low-Resource Domains via Meta-Learning

Cheonbok Park[1,*], **Yunwon Tae**[2,*], **Taehee Kim**[3], **Soyoung Yang**[3], **Mohammad Azam Khan**[4], **Lucy Park**[5,†], **and Jaegul Choo**[3]
[1]NAVER Corp,cbok.park@navercorp.com, [2]Korea University, tyj204@korea.ac.kr
[3]KAIST, {taeheekim, sy_yang, jchoo}@kaist.com
[4]DPDC, azam@dpdc.org.bd, [5]Upstage AI Research, lucy@upstage.ai

Abstract

Unsupervised machine translation, which utilizes unpaired monolingual corpora as training data, has achieved comparable performance against supervised machine translation. However, it still suffers from data-scarce domains. To address this issue, this paper presents a novel meta-learning algorithm for unsupervised neural machine translation (UNMT) that trains the model to adapt to another domain by utilizing only a small amount of training data. We assume that domain-general knowledge is a significant factor in handling data-scarce domains. Hence, we extend the meta-learning algorithm, which utilizes knowledge learned from high-resource domains, to boost the performance of low-resource UNMT. Our model surpasses a transfer learning-based approach by up to 2-3 BLEU scores. Extensive experimental results show that our proposed algorithm is pertinent for fast adaptation and consistently outperforms other baselines.

1 Introduction

Unsupervised neural machine translation (UNMT) leverages unpaired monolingual corpora for its training, without requiring an already labeled, parallel corpus. Recently, the state of the art in UNMT (Conneau and Lample, 2019; Song et al., 2019; Ren et al., 2019) has achieved comparable performances against supervised neural machine translation (NMT) approaches. In contrast to supervised NMT, which uses a parallel corpus, training the UNMT model requires a significant number of monolingual sentences (e.g., 1M-3M sentences). However, the prerequisite limits UNMT's applicability to low-resource domains, especially for domain-specific document translation tasks. Since gathering or creating those documents requires domain specific knowledge, the monolingual data themselves are scarce and expensive. In addition, the minority languages (e.g., Uzbek and Nepali) make the problem of data scarcity even worse.

Yet, UNMT for low-resource domains is not an actively explored field. One naive approach is to train a model on high-resource domains (e.g., economy and sports) while hoping the model will generalize on an unseen low-resource domain (e.g., medicine). However, recent studies have shown that non-trivial domain mismatch can significantly cause low translation accuracy on supervised NMT tasks (Koehn and Knowles, 2017).

Another reasonable approach is transfer learning—particularly, domain adaptation—which has shown performance improvements in the supervised NMT literature (Freitag and Al-Onaizan, 2016; Zeng et al., 2019). In this approach, the model is first pretrained using data from existing domains and then finetuned on a new domain. However, this approach can suffer from overfitting and catastrophic forgetting due to a small amount of training data and a large domain gap.

As an effective method for handling a small amount of training data, meta-learning has shown its superiority in various NLP studies such as dialog generation, machine translation, and natural language understanding (Qian and Yu, 2019; Gu et al., 2018; Dou et al., 2019). In general, the meta-learning approach is strongly affected by the number of different tasks where tasks are defined as languages or domains from the aforementioned studies. However, in practice, the previous studies may struggle to gather data to define tasks because they rely on a supervised model that requires labeled corpora. In this respect, we argue that applying a meta-learning approach to the unsupervised model is more feasible and achievable than the supervised

∗ equal contributions
† This work done in NAVER Corp.
Our code is avilable at https://github.com/papago-lab/MetaGUMT

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pages 2888–2901
August 1–6, 2021.

model because it can define multiple different tasks with unlabeled corpora. Therefore, we introduce a new meta-learning approach for UNMT, called MetaUMT, for low-resource domains by defining each task as a domain.

The objective of MetaUMT is to find the optimal initialization for the model parameters that can quickly adapt to a new domain even with only a small amount of monolingual data. As shown in Fig. 1 (a), we define two different training phases, a meta-train and a meta-test phase, and simulate the domain adaption process to obtain optimally initialized parameters. Specifically, the meta-train phase adapts model parameters to a domain while the meta-test phase optimizes the parameters obtained from the meta-train phase. After obtaining optimally initialized parameters through these two phases, we fine-tune the model using a target domain (i.e., a low-resource domain).

Although the initial parameters optimized through MetaUMT are suitable for adapting to a low-resource domain, these parameters may not fully maintain the knowledge of high-resource domains. Concretely, in the meta-test phase, MetaUMT optimizes initial parameters using the adapted parameters; however, it discards meta-train knowledge used to update adapted parameters in the meta-train phase. Therefore, instead of validating the same domain used in the meta-train phase, we intend to inject generalizable knowledge into the initial parameters by utilizing another domain in the meta-test phase. This prevents overfitting from the data scarcity issue.

As shown in Fig. 1 (b), we propose an improved meta-learning approach called MetaGUMT for low-resource UNMT by explicitly infusing common knowledge across multiple source domains as well as generalizable knowledge from one particular domain to another. In other words, we do not only encourage the model to find the optimally initialized parameters that can quickly adapt to a target domain with low-resource data, but also encourage the model to maintain common knowledge (e.g., general words such as determiners, conjunctions, and pronouns) which is obtainable from multiple source domains. Furthermore, due to a small amount of training data in a low-resource domain, the model can suffer from overfitting; however, we attempt to handle overfitting by leveraging generalizable knowledge that is available from one domain to another. Our proposed meta-learning approach demonstrates consistent improvements over other baseline models.

Overall, our contributions can be summarized as follows:

- We apply a meta-learning approach for UNMT. To the best of our knowledge, this is the first study to use a meta-learning approach for UNMT, where this approach is more suitable to a UNMT task than a supervised one.
- We empirically demonstrate that our enhanced method, MetaGUMT, shows fast convergence on both pre-training (i.e., meta-learning with source domains) and finetuning (i.e., adapting to a target domain).
- The model trained with MetaGUMT consistently outperforms all baseline models including MetaUMT. This demonstrates that finding optimally initialized parameters that incorporate high-resource domain knowledge and generalizable knowledge is significant in handling a low-resource domain.

2 Related Work

Our study leverages two components from the natural language processing (NLP) domain: low-resource NMT and meta-learning. In this section, we discuss previous studies by concentrating on these two main components.

2.1 Low-Resource Neural Machine Translation

Based on the success of attention-based models (Luong et al., 2015; Vaswani et al., 2017), NMT obtains significant improvement in numerous language datasets, even showing promising results (Wu et al.) in different datasets. However, the performance of NMT models depends on the size of the parallel dataset (Koehn and Knowles, 2017). To address this problem, one conventional approach is utilizing monolingual datasets.

Recent studies point out the difficulty of gathering parallel data, whereas the monolingual datasets are relatively easy to collect. To facilitate monolingual corpora, several studies apply dual learning (He et al., 2016), back-translation (Sennrich et al., 2016b), and pretraining the model with bilingual corpora (Hu et al., 2019; Wei et al., 2020). Furthermore, as a challenging scenario, recent studies propose the UNMT methods without using any

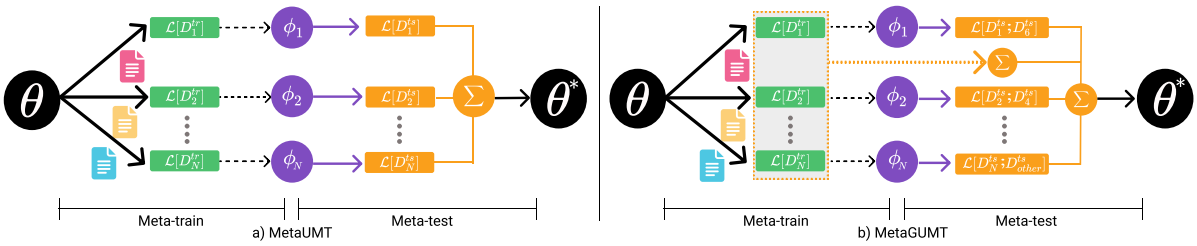

Figure 1: An illustration of a high-level training process for both MetaUMT and MetaGUMT. In the case of MetaGUMT, the training process is divided into two different phases, a meta-train phase and a meta-test phase. The objective in the meta-train phase is to obtain adapted parameters (i.e., ϕ) by minimizing a meta-train loss (i.e., $\mathcal{L}[D_N^{tr}]$) from initial unadapted parameters. N represents the number of domains; D^{tr} indicates meta-train data. In the meta-test phase, we optimize initial parameters θ through ϕ by minimizing the two losses, meta-train and meta-test losses, i.e., $\sum \mathcal{L}[D_N^{tr}]$ and $\sum \mathcal{L}[D_N^{ts}; D_{other}^{ts}]$. D^{ts} represents meta-test data; D_{other} is the domain data other than D_N.

parallel corpora (Lample et al., 2018a; Artetxe et al., 2018; Yang et al., 2018). The UNMT models show comparable performances by extending the back-translation method (Conneau et al., 2018) and incorporating methods such as shared Byte Pair Encoding (BPE) (Lample et al., 2018b) and cross-lingual representations (Conneau and Lample, 2019), following those of the supervised NMT. However, since these approaches require plenty of monolingual datasets, they suffer in a low-resource domain.

Transferring the knowledge from high-resource domains to a low-resource domain is one alternative way to address this challenge. A few studies concentrate on transferring the knowledge from the rich-resource corpora into the low-resource one. Several models (Chu and Wang, 2018; Hu et al., 2019) show better performances than when trained with the low-resource corpora only. However, these approaches are applicable in specific scenarios where one or both of the source and target domains consist of a parallel corpus.

To address the issues, we define a new task as the unsupervised domain adaptation on the low-resource dataset. Our work is more challenging than any other previous studies, since we assume that both the low-resource target domain and the source domain corpora are monolingual.

2.2 Meta Learning

Given a small amount of training data, most of the machine learning models are prone to overfitting, thus failing to find a generalizable solution. To handle this issue, meta-learning approaches seek for how to adapt quickly and accurately to a low-resource task, and show impressive results in various domains (Finn et al., 2017; Javed and White, 2019). The meta-learning approaches aim to find the optimal initialization of the model parameters that adapts the model to a low-resource dataset in a few iterations of training (Finn et al., 2017; Ravi and Larochelle, 2016). Owing to the success of the meta learning, recent studies apply the meta learning to low-resource NMT tasks, including multilingual NMT (Gu et al., 2018) and the domain adaptation (Li et al., 2020). These studies assume that all the training corpora consist of the parallel sentences. However, a recent work (Li et al., 2018) utilizes the meta learning approach to find a generalized model for multiple target tasks. However, it is not focused on adapting a specific target task since its main goal is to handle the target task without using any low-resource data.

Our study attempts to address the low-resource UNMT by exploiting meta-learning approaches. Moreover, we present two novel losses that encourage incorporating high-resource knowledge and generalizable knowledge into the model parameters. Our proposed approaches show significant performance improvements in adapting to a low-resource target domain.

3 Unsupervised Neural Machine Translation

In this section, we first introduce the notation of the general UNMT models. We then describe the three steps for the UNMT task: initialization, language modeling, and back-translation. On these three steps, we illustrate how each step contributes to improving the performance of UNMT.

Notations. We denote S and T as a source and a target monolingual language dataset. x and y represent the source and the target sentences from S and T. We assume the NMT model is parame-

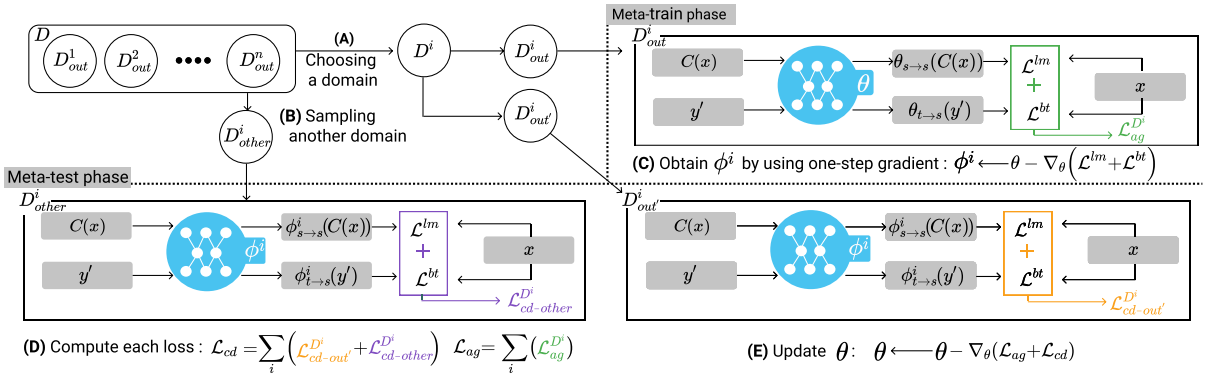

Figure 2: Overall training process of our proposed MetaGUMT. **(A)** A single domain (e.g., Law) is first chosen to compute $\mathcal{L}^{\mathcal{D}^i}_{ag}$ with model parameters θ in the meta-train phase and $\mathcal{L}^{\mathcal{D}^i}_{cd-out'}$ with temporary model parameters ϕ^i in the meta-test phase. **(B)** Another domain (e.g., IT) is sampled to compute $\mathcal{L}^{\mathcal{D}^i}_{cd-other}$ based on ϕ^i in the meta-test phase. **(C)** Temporary model parameters ϕ^i is updated from θ to learn the knowledge of high-resource domains. **(D)** Cross-domain and aggregated meta-train loss functions are computed across all out-domain datasets. **(E)** The optimal initialization θ is obtained by minimizing $\mathcal{L}_{ag}$ and $\mathcal{L}_{cd}$.

terized by θ. We also denote $M_{\text{s}\to\text{s}}$ and $M_{\text{t}\to\text{t}}$ as language models in a source and a target language, respectively, while denoting $M_{\text{s}\to\text{t}}$ and $M_{\text{t}\to\text{s}}$ as the machine translation models from the source to the target language and vice versa.

Initialization. A recent UNMT model (Lample et al., 2018b) is based on a shared encoder and decoder architecture for the source and the target language. Due to the shared encoder and decoder for each language, initializing the model parameters of the shared encoder and decoder is an important step for competitive performances (Conneau et al., 2018; Lample et al., 2018a; Artetxe et al., 2018; Yang et al., 2018). Conneau and Lample (2019) propose the XLM (cross-lingual language model) to initialize parameters, showing significantly improved performances for UNMT. Among various initialization methods, we leverage the XLM as our initialization method.

Language modeling. We use a denoising auto-encoder (Vincent et al., 2008) to train the UNMT model, reconstructing an original sentence from a noisy one in a given language. The objective function is defined as follows:

$$\begin{aligned}\mathcal{L}^{\text{lm}} =& \mathbb{E}_{x\sim S}[-\log M_{s\to s}(x|C(x))]+ \\ & \mathbb{E}_{y\sim T}[-\log M_{t\to t}(y|C(y))],\end{aligned} \quad (1)$$

where C is a noise function described in (Lample et al., 2018b), which randomly drops or swaps words in a given sentence. By reconstructing the sentence from the noisy sentence, the model learns the language modeling in each language.

Back-translation. Back-translation helps the model learn the mapping functions between the source and the target language by using only the monolingual sentences. For example, we sample a sentence x and y from source language S and target language T. To make pseudo-pair sentences from the sampled source sentence, we deduce the target sentence from the source sentence, such that $y' = M_{\text{s}\to\text{t}}(x)$, resulting in the pseudo parallel sentence, i.e., (x, y'). Similarly, we obtain (x', y), where x' is the translation of a target sentence, i.e., $M_{\text{t}\to\text{s}}(y)$. We do not back-propagate when we generate the pseudo-parallel sentence pairs. In short, the back-translation objective function is

$$\begin{aligned}\mathcal{L}^{\text{bt}} =& \mathbb{E}_{y\sim T}[-log M_{s\to t}\left(y \mid x'\right)]+ \\ & \mathbb{E}_{x\sim S}[-log M_{t\to s}\left(x \mid y'\right)].\end{aligned} \quad (2)$$

4 Proposed Approach

This section first explains our formulation of a low-resource unsupervised machine translation task where we can apply a meta-learning approach. Afterwards, we elaborate our proposed methods, MetaUMT and MetaGUMT. We utilize the meta-learning approach to address a low-resource challenge for unsupervised machine translation. Moreover, we extend MetaUMT into MetaGUMT to explicitly incorporate learned knowledge from multiple domains.

4.1 Problem Setup

Finn et al. (2017) assume multiple different tasks to find the proper initial parameters that can quickly adapt to a new task using only a few training

examples. In this paper, we consider tasks in the meta-learning as domains, where $\mathcal{D}_{out} = \{\mathcal{D}^0_{out}, ..., \mathcal{D}^n_{out}\}$ represents n out-domain datasets (i.e., source domain datasets), and $\mathcal{D}_{in}$ indicates an in-domain dataset (i.e., a target domain dataset), which can be the dataset in an arbitrary domain not included in $\mathcal{D}_{out}$. Each domain in both $\mathcal{D}_{out}$ and $\mathcal{D}_{in}$ is assumed to be composed of unpaired language corpora, and we create $\mathcal{D}_{in}$ as a low-resource monolingual dataset [1]. To adapt our model to the low-resource in-domain data, we finetune the UNMT model by minimizing both the losses described in Eqs. (1) and (2) with $\mathcal{D}_{in}$.

4.2 MetaUMT

In order to obtain an optimal initialization of the model parameters, allowing the model to quickly adapt to a new domain with only a small number of monolingual training data, MetaUMT uses two training phases, the *meta-train* phase and the *meta-test* phase. During the meta-train phase, the model first learns domain-specific knowledge by updating initial model parameters θ to temporary model parameters ϕ^i, i.e., adapted parameters. Then, in the meta-test phase, the model learns the adaptation by optimizing θ with respect to ϕ^i. From the domain adaption perspective, two phases simulate the domain adaption process. The model first adapts to a specific domain through the meta-train phase, and this adaption is evaluated in the meta-test phase.

Meta-train phase. We obtain ϕ^i for each i-th out-domain dataset by using one-step gradient descent, i.e.,

$$\phi^i = \theta - \alpha \nabla_\theta \mathcal{L}^{\mathrm{s}}_{\mathcal{D}^{\mathrm{i}}_{out}}(\theta), \tag{3}$$

where $\mathcal{L}^{s}_{\mathcal{D}^{\mathrm{i}}_{out}}$ is represented as

$$\mathcal{L}^{s}_{\mathcal{D}^{\mathrm{i}}_{out}} = \mathcal{L}^{\mathrm{lm}}_{\mathcal{D}^{\mathrm{i}}_{out}}(\theta) + \mathcal{L}^{\mathrm{bt}}_{\mathcal{D}^{\mathrm{i}}_{out}}(\theta). \tag{4}$$

$\mathcal{D}^i_{out}$ is the i-th out-domain dataset, and α is the learning rate for the meta-train phase. As previously discussed in Section 3, the language modeling and back-translation losses are essential in facilitating the unsupervised machine translation. Hence, $\mathcal{L}^s$ consists of $\mathcal{L}^{\mathrm{lm}}$ and $\mathcal{L}^{\mathrm{bt}}$, where each loss function is computed with $\mathcal{D}^i_{out}$.

Meta-test phase. The objective of the meta-test phase is to update θ using each ϕ^i learned from the meta-train phase by using each $\mathcal{L}^{\mathrm{s}}_{\mathcal{D}^i_{out'}}$ [2]. We call this update as a meta-update, defined as

$$\theta \leftarrow \theta - \beta \nabla_\theta \sum_{i=0}^{n} \mathcal{L}^{\mathrm{s}}_{\mathcal{D}^i_{out'}}(\phi^i), \tag{5}$$

where β is another learning rate in the meta-test phase. Since Eq. (5) requires the second-order gradient, the equation is simplified with the first-order gradient by replacing the second-order term. Finn et al. (2017) showed that the first-order approximation of the meta-learning maintains the performance while minimizing the computational cost.

4.3 MetaGUMT

To handle a data scarcity issue from a meta-learning perspective, it is critical to be able to make the initialized model to adapt to a data-scarce domain. However, since a small amount of training data in the new domain may cause the model to overfit and prevent utilizing high-resource domain knowledge, it is important to incorporate high-resource domain knowledge and generalizable knowledge into the model parameters. To address this issue, we extend the existing meta-learning approach via two novel losses, which we call an aggregated meta-train loss and a cross-domain loss. The former contributes to incorporating high-resource domain knowledge into the model parameters, while the latter encourages our model, after trained using a particular domain, to still generalize well to another domain, i.e., cross-domain generalization.

Meta-train phase. As shown in Fig. 2 (C), via Eqs. (3) and (4), we obtain ϕ^i from each i-th out-domain datasets. Since this phase is exactly same with the meta-train phase of MetaUMT, we leave out the details.

Meta-test phase. The aggregated meta-train loss, which refers to Fig. 2 (D), is computed using all out-domain datasets, i.e.,

$$\mathcal{L}_{ag} = \sum_{i=0}^{n} \mathcal{L}^{\mathrm{s}}_{\mathcal{D}^i_{out}}(\theta). \tag{6}$$

This loss term allows the model to learn the source domain knowledge that is potentially applicable to a target domain. Moreover, to alleviate the overfitting after adapting to the low-resource domain, we

[1] We randomly sample the 5,000 tokens ($\sim$ 300 sentences) from the in-domain training dataset.

[2] $\mathcal{L}^{\mathrm{s}}_{\mathcal{D}^i_{out}}$ and $\mathcal{L}^{\mathrm{s}}_{\mathcal{D}^i_{out'}}$ indicate different batch sampled data from same $\mathcal{D}^i$.

Model	$\mathcal{D}_{out}$	Medical Koran	Law IT	EUB GV	Medical Koran	Law IT	EUB GV	Subtitles Europarl	Law IT	EUB GV	GV Subtitles	Europarl Medical	EUB Koran
	$\mathcal{D}_{in}$	Subtitles			Europarl			Medical			Law		
		De-En	En-De	epoch	De-En	En-De	epoch	De-En	En-De	epoch	De-En	En-De	epoch
Unadapted		9.46	7.54	-	22.31	15.82	-	21.30	19.23	-	31.1	25.35	-
Transfer		10.92	9.18	4	22.96	16.78	3	22.77	19.78	6	31.69	25.59	4
Mixed		11.77	9.96	15	22.99	17.05	5	22.98	19.99	8	31.69	25.74	6
MetaUMT		12.95	10.58	3	24.53	18.59	**2**	24.6	21.86	**4**	32.51	27.22	3
MetaGUMT		**13.45**	**10.89**	**2**	**25.13**	**18.95**	**2**	**25.32**	**22.79**	**4**	**34.26**	**29.37**	**2**
Supervised NMT		2.24	2.49	8	1.88	1.52	7	7.71	9.80	11	11.29	10.07	13
Unsupervised NMT		1.26	0.94	5	1.53	0.76	23	3.37	2.72	9	6.07	4.73	11

Table 1: BLEU scores on various out-domain (D_{out}) and in-domain (D_{in}) combinations for the language pairs of De-En and En-De. The "epoch" column indicates the converged number of epochs for each in-domain dataset. Since the unadapted model does not have any additional finetuning step, we leave the epoch column as blank. The bold represents the significant difference ($p < 0.05$) with others. Each BLEU score represents the average of ten trials.

introduce a cross-domain loss, which is in Fig. 2 (D), as

$$\mathcal{L}_{cd} = \sum_{i=0}^{n} \mathcal{L}^{s}_{\mathcal{D}^i_{cd}}(\phi^i), \tag{7}$$

where $\mathcal{L}^{s}_{\mathcal{D}^i_{cd}} = \mathcal{L}^{s}_{\mathcal{D}^i_{out'}}(\phi^i) + \mathcal{L}^{s}_{\mathcal{D}^i_{other}}(\phi^i)$, i.e., computing the cross-domain loss with the data from $\mathcal{D}^i_{out'}$ as well as those from other domains $\mathcal{D}^i_{other}$.

To obtain the optimal initialization θ for model parameters, we define our total loss function, which is Fig. 2 (E), as the sum of the two of our losses, i.e.,

$$\theta \leftarrow \theta - \beta \nabla_\theta (\mathcal{L}_{cd} + \mathcal{L}_{ag}). \tag{8}$$

In summary, our aggregated meta-train and cross-domain losses encourage our model to accurately and quickly adapt to an unseen target domain. The overall procedure is described in Algorithm A.1.

5 Experiments

This section first introduces experiment settings and training details. Afterwards, we show empirical results in various scenarios.

5.1 Dataset and Preprocessing

We conduct our experiments on eight different domains [3](Appendix T.2). Each domain dataset is publicly available on OPUS [4] (Tiedemann, 2012). We utilize the eight domains for out-domain (D_{out}) and in-domain datasets (D_{in}). To build the monolingual corpora of in-domain and out-domain datasets, we sample data from the parallel corpus. We made sure to include at most one sentence from each pair of parallel sentences. For instance, we sample the first half of the sentences as unpaired source data and the other half as truly unpaired target data. Consequently, the sampled monolingual corpora contain no translated sentence in each language. Each of the two monolingual corpora contains the equal number of sentences for each language (e.g., English and German). For our low-resource scenarios, we sample 5,000 tokens from a selected in-domain corpus for each language. Note that the out-domain dataset represents the full monolingual corpora.

5.2 Experimental Settings

As our base model, we use a Transformer (Vaswani et al., 2017), which is initialized by a masked language model from XLM (Conneau and Lample, 2019) using our out-domain datasets. All the models consist of 6 layers, 1,024 units, and 8 heads.

We establish and evaluate various baseline models as follows:

- **UNMT model** is trained with only the in-domain monolingual data, composed of 5,000 words for each language.
- **Supervised neural machine translation model (NMT)** is trained with in-domain parallel datasets, which we arrange in parallel with the two in-domain monolingual corpora.
- **Unadapted model** is pretrained with only the out-domain datasets and evaluated on the in-domain datasets.
- **Transfer learning model** is a finetuned model, which is pretrained with the out-domain datasets and then finetuned with a low-resource in-domain dataset.
- **Mixed finetuned model** (Chu et al., 2017) is similar to the *transfer learning model*, but

[3]Acquis (Law), EMEA (Medical), IT, Tanzil (Koran), Subtitles, EUbookshop (EUB), Europarl, and GlobalVoices (GV)

[4]http://opus.nlpl.eu/

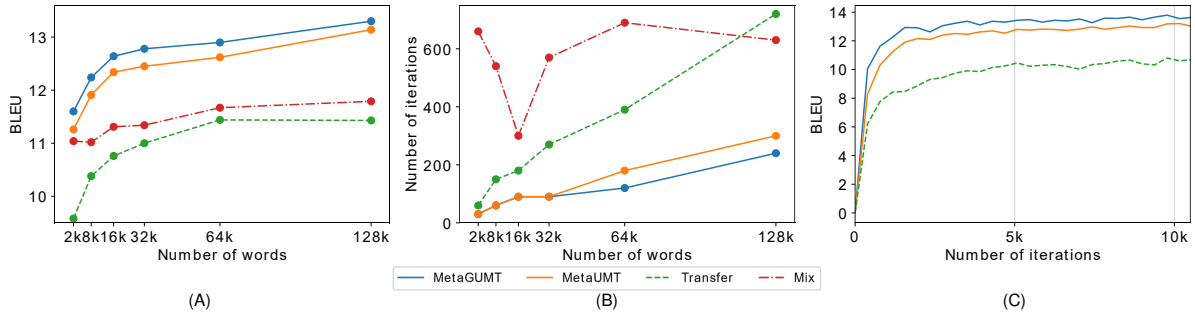

Figure 3: Results of the models that are first pretrained on Medical, Law, EUbookshop, Koran, IT, and GlobalVoices datasets and then finetuned on a Subtitles dataset. (A) is a performance comparison with respect to the number of words for adaptation. (B) is the number of iterations until the convergence during the finetuning stage with respect to the number of words. (C) is the number of iterations until convergence, where the BLEU is validating scores calculated by the average of En-De and De-En.

it utilizes both in-domain and out-domain datasets for finetuning. That is, the training batch is sampled evenly from in-domain and out-of-domain datasets.

5.3 Experimental Results

In order to verify that leveraging the high-resource domains (i.e., the source domains) effects to handle the low-resource domains (i.e., the target domain), we compare the unsupervised and supervised models with ours and other baseline models.

As shown in Table 1, the unsupervised model trained on in-domain data suffers from data scarcity because it only uses low-resource in-domain data. Although the unsupervised and supervised models are initialized by XLM, those models show the worst performance in all the cases. This result indicates that when the small size of an in-domain corpus is given, it is appropriate to utilize the out-domain datasets rather than to train only with low-resource data. In addition, the performance of the unadapted model is far behind compared to other models, such as the mixed finetuned model, transfer learning model, MetaUMT, and MetaGUMT. This implies that we need an adequate strategy of leveraging the high-resource domains to improve the performance.

We further compare the performance between our proposed approaches (i.e., MetaUMT and MetaGUMT) and the other two finetuning models (i.e., the transfer learning and the mixed finetuned ones). Our methods exhibit the leading performances in both directions of translation ($en \leftrightarrow de$), and consistently achieve improvements of 2-3 BLEU score in most of settings. Furthermore, MetaGUMT consistently obtains better BLEU scores and converges faster than MetaUMT. We assert that our proposed losses (i.e., the aggregated meta-train and the cross-domain losses) help the model not only to perform well even on the unseen in-domain dataset but also to accelerate the convergence speed.

5.4 Performances and Adaptation Speed in Finetuning Stage

As shown in Fig. 3 (A), we compare our proposed methods with the transfer learning approach by varying the sizes of an in-domain monolingual corpus. The smaller the size of training data is, the wider the performance gap between the two approaches and the transfer learning model becomes. It means that meta-learning is an effective approach to alleviate the performance degradation, preventing the model from overfitting to the low-resource data.

Compared to the transfer learning model, MetaUMT demonstrates a better performance than other methods in various settings. However, MetaGUMT exhibits even better performances consistently in all settings owing to our proposed losses (Eq. (8)). The transfer learning approach shows the worst performance except for the unadapted model, even though it exploits the in-domain corpus after being pretrained with the out-domain datasets.

Additionally, we analyze the number of iterations required for a model to converge given an in-domain dataset. As shown in Fig. 3 (B), the meta-learning approaches rapidly converge after only a few iterations, even faster than the transfer learning one does. As the number of in-domain training words increases, the transfer learning approach requires a much larger number of iterations until convergence than our meta-learning approaches. It can be seen that MetaUMT and MetaGUMT rapidly adapt to an unseen domain. Moreover, owing to the encapsulated knowledge from the high-resource do-

Parameter	Initial θ												Finetuned θ			
D	$\mathcal{D}_{out}$												$\mathcal{D}_{in}$		Unseen	
	Meidcal		Law		Koran		EUB		IT		GV		Subtitles		Europarl	
	De-En	En-De	De-En	En-De	De-En	En-De	De-En	En-De	De-En	En-De	De-En	En-De	De-En	En-De	De-En	En-De
Transfer	30.98	26.96	34.8	30.28	13.72	11.59	12.32	10.01	20.98	17.74	17.4	14.25	10.92	9.18	22.31	16.58
Mixed finetuned	-	-	-	-	-	-	-	-	-	-	-	-	11.77	9.96	22.84	16.92
MetaUMT	33.0	23.39	27..74	15.4	4.89	0.79	6.78	2.59	9.45	4.68	2.77	1.06	12.95	10.58	23.91	18.7
MetaGUMT	**37.37**	**31.63**	**42.73**	**37.3**	**18.2**	**13.84**	**13.72**	**11.8**	**24.0**	**19.24**	**21.24**	**17.38**	**13.45**	**10.89**	**24.44**	**19.31**

Table 2: BLEU scores evaluated on out-domain and in-domain data with initial θ and finetuned θ, respectively. "$\mathcal{D}$" denotes the domain, "Unseen" indicates the new domain evaluated with finetuned θ. Since the transfer and mixed finetuned models use the same initial θ, we leave its corresponding row as "-".

mains, MetaGUMT converges within a relatively earlier iteration than MetaUMT does.

In summary, the meta-learning-based methods quickly converge in the low-resource domain, improving the performances over the transfer learning method in various low-resource settings. This indicates that the meta-learning-based approaches are suitable to alleviate the data deficiency issue in scarce domains. Furthermore, our losses in Eq. (8) enhance the capabilities of aggregating domain general knowledge and finding adequate initialization.

5.5 Number of Iterations until Convergence in Pretraining Stage

An advantage of our meta-learning approaches is that they can find an optimal initialization point from which the model can quickly adapt to a low-resource in-domain dataset. The transfer learning model requires twice more iterations until convergence than ours does. As shown in Fig. 3 (C), MetaUMT and MetaGUMT not only converge quickly but also outperform the other baseline methods. Specifically, compared to MetaUMT, MetaGUMT is effective in achieving an optimized initialization at an earlier iteration. These results indicate that our additional losses (i.e., the cross-domain and aggregated meta-train losses) are beneficial in boosting up the ability for finding an optimal initialization point when training the model with the out-domain datasets.

5.6 Analysis of MetaGUMT losses

We assume that the domain generalization ability and high-resource domain knowledge are helpful for the UNMT model to translate the low-resource domain sentences. First, to identify whether the model encapsulates the high-resource knowledge from multiple sources, we evaluate our model on out-domain datasets (i.e., D_{out}) with initial θ. As shown in Table. 2, MetaGUMT shows remarkable performances over MetaUMT in all domains, even better than the transfer learning models. In other words, MetaUMT demonstrates poor performances

Cross-domain	Aggregated meta-train	De-En	En-De	Average	Δ
✗	✗	27.09	24.6	25.85	
✓	✗	27.37	24.76	26.06	+0.21
✗	✓	27.54	24.90	26.22	+0.37
✓	✓	**27.85**	**25.06**	**26.46**	+0.61

Table 3: Effectiveness of each cross-domain and aggregated meta-train loss.

in D_{out}, compared to MetaGUMT. This can be explained as MetaGUMT uses an aggregated meta-train loss such that MetaGUMT is able to encapsulate the high-resource domain knowledge. As shown in Table. 1, MetaGUMT achieves superior performances, showing that MetaGUMT is capable of leveraging the encapsulated knowledge when finetuning the low-resource target domain.

Secondly, our cross-domain loss encourages the model to have a generalization capability after adapting to the low-resource target domain. As shown in "Unseen" column in Table. 2, MetaGUMT outperforms the other models. It can be seen that our model has the domain generalization ability after the finetuning stage due to the cross-domain loss in the meta-test phase.

5.7 Performance of Unbalanced Monolingual Data in Finetuing Stage

In UNMT, data unbalancing is often the case in that source language (e.g., English) data are abundant and the target language (e.g., Nepali) data are scarce (Kim et al., 2020). We extend our experiment to the unbalanced scenarios to examine whether our proposed model shows the same tendency. In this scenario, the low-resource target domain dataset consists of monolingual sentences from one side with two times more tokens than the monolingual sentences from the other. As shown in Table. 4, MetaGUMT outperforms in all unbalanced data cases. It shows that MetaGUMT is feasible to a practical UNMT scenario where the number of sentences is different in the source and target languages. The only difference against the main experiment setting 5.1 is the condition that the in-domain corpus is unbalanced. We also include the

# tokens		Mixed		MetaUMT		MetaGUMT	
En	De	En-De	De-En	En-De	De-En	En-De	De-En
5k	10k	26.04	31.90	28.80	32.65	**29.43**	**34.28**
8k	16k	26.09	32.01	27.84	32.93	**29.62**	**34.39**
16k	32k	26.44	32.37	27.92	32.96	**30.10**	**34.44**
32k	64k	27.39	32.84	28.67	33.52	**29.83**	**34.77**

Table 4: Results on the unbalanced monolingual Law domain data during the finetuning stage, where D_{out} is GV, Euorparl, EUB, Subtitles, Medical and Koran.

result of the transfer learning model in Table. T.4.

5.8 Ablation Study

We empirically show the effectiveness of the cross-domain and aggregated meta-train losses, as shown in Table 3 [5]. First, compared to MetaUMT which does not use any of the two losses, incorporating the cross-domain loss improves the average BLEU score by 0.21. The cross-domain loss acts as a regularization function that prevents the model from overfitting during the finetuning stage. Second, the aggregated meta-train loss, another critical component of our model, allows the model to utilize the high-resource domain knowledge in the finetuning stage. This also improves the average BLEU score by 0.37 from MetaUMT. Lastly, combining both cross-domain and aggregated meta-train losses significantly enhances the result in both directions of translation ($En \leftrightarrow De$), indicating that they are complementary to each other.

5.9 Impact of the Number of Source Domains

We examine how the performances change against the different number of source domains for each approach. As shown in Table. 5 [6], MetaGUMT consistently outperforms the transfer, the mixed-finetune, and MetaUMT approaches. As the size of the source domains increases, so does the performance gap between ours and the transferring based models, i.e., transferring and mixed-finetune models. This indicates that the meta-learning based approaches are highly effected by the size of the domains in the meta-train phase, and also, if the number of source domains is large enough to capture the general knowledge, the meta-learning based approaches are suitable to handle the low-resource target task (i.e., machine translation in a low-resource domain).

[5]The models are pretrained on Subtitles, Law, EUB, Europarl, IT, and GV and then finetuned on the Medical data.

[6]The *4* case contains the Medical, Law, Koran and EUB domains. *5* and *6* additionally utilize one more domain(i.e., IT) and two more domains(i.e.,IT and GV), respectively.

# $\mathcal{D}_{out}$	MetaGUMT		MetaUMT		Transfer		Mixed	
	En-De	De-En	En-De	De-En	En-De	De-En	En-De	De-En
4	**5.97**	**7.47**	5.87	7.24	5.75	7.17	5.87	7.22
5	**7.58**	**9.49**	7.33	9.01	7.17	8.08	7.20	8.68
6	**10.89**	**13.45**	10.58	12.95	9.18	10.92	9.96	11.77

Table 5: Effectiveness of the different number of source domains between meta-learning based approaches and the transfer learning approach, where # $\mathcal{D}_{out}$ represents the number of out-domain datasets in the pretraining stage.

6 Conclusions

This paper proposes a novel meta-learning approach for low-resource UNMT, called MetaUMT, which leverages multiple source domains to quickly and effectively adapt the model to the target domain even with a small amount of training data. Moreover, we introduce an improved method called MetaGUMT, which enhances cross-domain generalization and maintains high-resource domain knowledge. We empirically show that our proposed approach consistently outperforms the baseline methods with a nontrivial margin. We believe that our proposed methods can be extended to semi-supervised machine translation as well. In the future, we will further analyze other languages, such as Uzbek and Nepali, instead of languages like English and German.

Acknowledgments

This work was partially supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2019-0-00075, Artificial Intelligence Graduate School Program (KAIST) and No. 2020-0-00368, A Neural-Symbolic Model for Knowledge Acquisition and Inference Techniques) and by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (No. NRF-2019R1A2C4070420)

References

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2018. Unsupervised neural machine translation. *Proc. International Conference on Learning Representations (ICLR)*. Vancouver, Canada.

Chenhui Chu, Raj Dabre, and Sadao Kurohashi. 2017. An empirical comparison of domain adaptation methods for neural machine translation. In *In Proc. the Annual Meeting of the Association for Computational Linguistics (ACL) (Volume 2: Short Papers)*, pages 385–391, Vancouver, Canada.

Chenhui Chu and Rui Wang. 2018. A survey of domain adaptation for neural machine translation. In *Proc. the International Conference on Computational Linguistics (COLING)*, pages 1304–1319, Santa Fe, New Mexico, USA.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, pages 7057–7067, Vancouver, Canada.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *Proc. International Conference on Learning Representations (ICLR)*. Vancouver, Canada.

Zi-Yi Dou, Keyi Yu, and Antonios Anastasopoulos. 2019. Investigating meta-learning algorithms for low-resource natural language understanding tasks. In *Proc. the Conference on Empirical Methods in Natural Language Processing and the International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1192–1197, Hong Kong, China.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proc. the International Conference on Machine Learning (ICML)*, volume 70, page 1126–1135.

Markus Freitag and Yaser Al-Onaizan. 2016. Fast domain adaptation for neural machine translation. *CoRR*, abs/1612.06897.

Jiatao Gu, Yong Wang, Yun Chen, Victor O. K. Li, and Kyunghyun Cho. 2018. Meta-learning for low-resource neural machine translation. In *Proc. the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3622–3631, Brussels, Belgium.

Di He, Yingce Xia, Tao Qin, Liwei Wang, Nenghai Yu, Tie-Yan Liu, and Wei-Ying Ma. 2016. Dual learning for machine translation. In *Proc. Advances in Neural Information Processing Systems (NIPS)*, volume 29.

Junjie Hu, Mengzhou Xia, Graham Neubig, and Jaime Carbonell. 2019. Domain adaptation of neural machine translation by lexicon induction. In *Proc. the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2989–3001, Florence, Italy.

Khurram Javed and Martha White. 2019. Meta-learning representations for continual learning. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, Vancouver, Canada.

Yunsu Kim, Miguel Graça, and Hermann Ney. 2020. When and why is unsupervised neural machine translation useless? In *Proc. the Annual Conference of the European Association for Machine Translation (EACL)*, pages 35–44, Lisboa, Portugal.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proc. the Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proc. of the First Workshop on Neural Machine Translation (WMT)*, pages 28–39, Vancouver, Canada.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018a. Unsupervised machine translation using monolingual corpora only. *International Conference on Learning Representations*. Vancouver, Canada.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018b. Phrase-based & neural unsupervised machine translation. In *Proc. the Conference on Empirical Methods in Natural Language Processing*, pages 5039–5049, Brussels, Belgium.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy Hospedales. 2018. Learning to generalize: Meta-learning for domain generalization. In *Proc.the AAAI Conference on Artificial Intelligence (AAAI)*, volume 32.

Rumeng Li, Xun Wang, and Hong Yu. 2020. MetaMT, a meta learning method leveraging multiple domain data for low resource machine translation. In *Proc. the AAAI Conference on Artificial Intelligence (AAAI)*, volume 34, pages 8245–8252.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proc. the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1412–1421, Lisbon, Portugal.

Kun Qian and Zhou Yu. 2019. Domain adaptive dialog generation via meta learning. In *Proc. the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2639–2649, Florence, Italy.

Sachin Ravi and Hugo Larochelle. 2016. Optimization as a model for few-shot learning. *Proc. International Conenference on Learning Representations (ICLR)*. Vancouver, Canada.

Shuo Ren, Yu Wu, Shujie Liu, Ming Zhou, and Shuai Ma. 2019. Explicit cross-lingual pre-training for unsupervised machine translation. pages 770–779.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *Proc. the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 86–96, Berlin, Germany.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *Proc. the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1715–1725, Berlin, Germany.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. MASS: Masked sequence to sequence pre-training for language generation. In *Proc. the International Conference on Machine Learning (ICML)*, volume 97, pages 5926–5936.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proc. the International Conference on Language Resources and Evaluation (LREC)*, pages 2214–2218, Istanbul, Turkey.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proc. Advances in Neural Information Processing Systems (NIPS)*, pages 5998–6008, Long Beach, CA, USA.

Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. 2008. Extracting and composing robust features with denoising autoencoders. In *Proc. of the International Conference on Machine Learning (ICML)*, pages 1096–1103, Helsinki, Finland.

Hao-Ran Wei, Zhirui Zhang, Boxing Chen, and Weihua Luo. 2020. Iterative domain-repaired back-translation. In *Proc. the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5884–5893, Online. Association for Computational Linguistics.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144.

Zhen Yang, Wei Chen, Feng Wang, and Bo Xu. 2018. Unsupervised neural machine translation with weight sharing. In *Proc. the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 46–55, Melbourne, Australia.

Jiali Zeng, Yang Liu, Jinsong Su, Yubing Ge, Yaojie Lu, Yongjing Yin, and Jiebo Luo. 2019. Iterative dual domain adaptation for neural machine translation. In *Proc the Conference on Empirical Methods in Natural Language Processing and the International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 845–855, Hong Kong, China.

A Implementation Details

In order to preprocess datasets, We utilize Moses (Koehn et al., 2007) to tokenize the sentences. We then use byte-pair encoding (BPE) (Sennrich et al., 2016a) to build a shared sub-word vocabulary using fastBPE[7] with 60,000 BPE codes. Based on this shared sub-word vocabulary, constructed from the out-domain datasets, we split words into sub-word units for the in-domain dataset. We implement all of the models using PyTorch library [8], and then train them in four nvidia V100 gpus for pretraining and finetuning. We evaluate all the experiments based on the BLEU script [9]. The number of convergence iteration of each algorithm is defined based on the best validation epoch, which shows no more improvement on validation score after we run 10 more epochs. Moreover, we have conducted comprehensive experiments to obtain our main result table (Table. 1 and Table. T.1) on different domains by training the model with 10 different sampled words each time.

For optimizing each algorithms, we choose the Adam optimizer (Kingma and Ba) for pretraining stage, as well as the Adam warmup optimizer (Vaswani et al., 2017) for finetuning stage. The learning rate is set to 10^{-4}, optimized within the range of 10^{-2} to 10^{-5}. In all experiments, the number of tokens per batch is set as 1,120 and the dropout rate is set as 0.1. In meta-learning approaches, we set the learning rates of alpha and beta commonly as 0.0001 in all experiments.

In the pretraining stage, we follow the same stopping criterion as Gu et al. (2018). For instance, among different target domains, we randomly select one as a validation domain. We utilize early stopping, i.e., stopping training if the validation BLEU score does not increase within the ten subsequent epochs. Similarly in the finetuning stage, we apply early stopping using a validation dataset from the target domain.

B Additional Results on Different Domain Combinations

Since the combination of D_{out} and D_{in} can consist differently, this section provides additional results. As shown in Table. T.1, our proposed approaches still significantly outperform other baseline models in different domain combination settings.

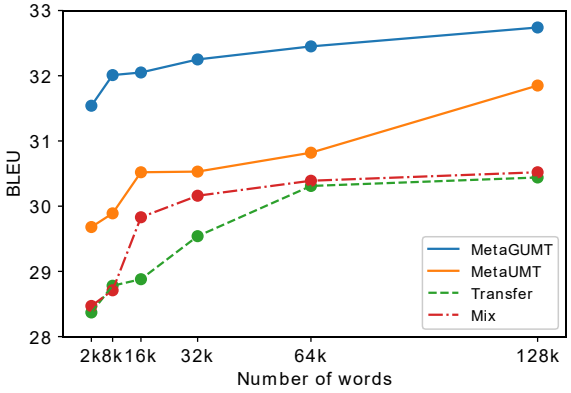

Figure C.1: A performance comparison with respect to the number of words for adaptation on a Law domain.

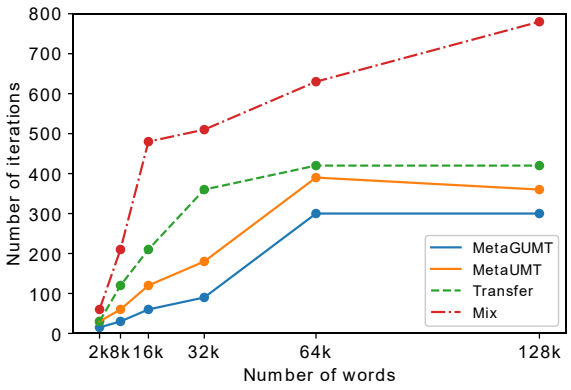

Figure C.2: Number of iterations until the convergence during the finetuining stage with respect to the number of words on a Law domain.

C Perfomances and Adaptation Speed in Finetuning Stage for a Law Domain

As shown in Fig C.1 and Fig C.2, MetaGUMT consistently outperforms other methods even though the number words are increasing. Through this experiment, we attempt to show the robustness of our methods (i.e., MetaUMT and MetaGUMT) against others (i.e., transferring and mixed-finetune models). The models are pretrained on Subtitles, EUbookshop, Europarl, GlobalVoices, Medical, and Koran datasets and then finetuned on a Law dataset.

D Comparison between MetaGUMT and MetaUMT Algorithms

As shown in Algorithms. A.1, we provide an overall algorithm of MetaGUMT. The only difference

[7] https://github.com/glample/fastBPE

[8] https://pytorch.org/

[9] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/generic/multi-bleu.perl

Model	$\mathcal{D}_{out}$	Medical Subtitles	Law EUB	Koran Europarl	Medical Subtitles	Law EUB	Koran Europarl	GV Subtitles	Europarl Medical	EUB Koran
	$\mathcal{D}_{in}$	IT			GV			IT		
		De-En	En-De	epoch	De-En	En-De	epoch	De-En	En-De	epoch
Unadapted		18.62	14.89	-	19.27	16.65	-	16.10	15.30	-
Transfer		19.80	16.35	4	19.99	16.90	3	19.31	16.13	5
Mixed		19.75	16.49	7	20.03	16.95	5	19.39	16.18	8
MetaUMT		21.08	18.05	4	22.36	18.91	3	20.5	17.06	**4**
MetaGUMT		**21.37**	**18.42**	**3**	**22.76**	**19.24**	**2**	**20.74**	**17.74**	**4**
Supervised NMT		3.48	3.33	15	0.97	0.85	14	3.53	3.59	10
Unsupervised NMT		1.83	0.86	22	0.51	0.18	20	0.51	0.55	7

Table T.1: Extended results on various domain settings. The column 'epoch' indicates the converged number of epochs for each in-domain dataset. Since the unadapted model does not involve an additional finetuning step, we leave the epoch column as blank.

Algorithm A.1 MetaGUMT

Require: α, β: step sizes

1: Pretrain θ by using XLM
2: **while** not done **do**
3: **for** all $\mathcal{D}^i_{out}$ **do**
4: Evaluate $\nabla_\theta L^{\text{lm}}_{\mathcal{D}^i_{out}}(\theta)$ with respect to source and target language sentences from $\mathcal{D}^i_{out}$
5: **Back-translation** generates source and target language sentences using the current translation model
6: Evaluate $\nabla_\theta \mathcal{L}^{\text{bt}}_{\mathcal{D}^i_{out}}(\theta)$ with using pseudo-generated sentences
7: Sum each gradient: $\nabla_\theta \mathcal{L}^{\text{s}}_{\mathcal{D}^i_{out}} = \nabla_\theta \mathcal{L}^{\text{lm}}_{\mathcal{D}^i_{out}}(\theta) + \nabla_\theta \mathcal{L}^{\text{bt}}_{\mathcal{D}^i_{out}}(\theta)$
8: Compute adapted parameters with one-step gradient descent: $\phi^i = \theta - \alpha \nabla_\theta \mathcal{L}^{\text{s}}_{D^i_{out}}(\theta)$
9: **end for**
10: Update $\theta \leftarrow \theta - \beta \nabla_\theta (\mathcal{L}_{cd} + \mathcal{L}_{ag})$
11: **end while**

between MetaUMT and MetaGUMT is the meta-test phase in line 10. While MetaUMT computes the loss using Eq. (5), MetaGUMT utilizes Eq. (8).

E Performance of Semi-Superivsed Machine Translation in Finetuning Stage

The proposed algorithms, MetaUMT and MetaGUMT, show promising results on low-resource monolingual data. However, some may argue that creating parallel sentences from a small number of unpaired monolingual sentences (e.g., 5k tokens) is also feasible. Hence, we additionally conduct an experiment of semi-supervised machine translation in the finetuning stage. For instance, we follow the same pretraining stage, but we utilize both monolingual and parallel sentences while finetuning the model on a low-resource domain. The number of tokens for each monolingual and parallel data is 5k. To finetune the model in the semi-supervised setting, we compute the loss as sum of $\mathcal{L}^{ct}$ and $\mathcal{L}^{bt}$, where $\mathcal{L}^{ct}$ is the conventional translation loss in the supervised NMT, i.e.,

$$\begin{aligned}\mathcal{L}^{\text{ct}} =& \mathbb{E}_{(x,y)\sim P}[-log M_{s\to t}\left(y \mid x\right)] + \\ & \mathbb{E}_{(x,y)\sim P}[-log M_{t\to s}\left(x \mid y\right)].\end{aligned} \tag{9}$$

As shown in Table T.3, we observe that MetaGUMT demonstrates the promising performance against others, even if we only utilize the monolingual out-domain datasets to pretrain the model.

Corpus	Words		Sentences	W/S	
	EN	DE		EN	DE
Acquis (Law)	9.2M	8M	0.7M	12.93	11.30
EMEA (Medical)	7.5M	6.3M	1.1M	6.81	5.75
IT	1.7M	1M	0.3M	9.08	5.32
Tanzil (Koran)	5.6M	5.3MS	0.5M	10.66	10.08
Subtitles	92.7M	87.6M	22.5M	4.11	3.89
EUbookshop (EUB)	115.4M	100M	9.3M	12.37	10.72
Europarl	27.3M	25.7M	1.9M	13.99	13.18
GlobalVoices (GV)	0.6M	0.6M	0.05M	10.67	10.88

Table T.2: Statistics of each corpora.

F Statistics of Datasets

As shown in Table. T.2, we present the overall number of sentences and words for each domain, where W/S indicates the number of words per sentence in a domain.

# tokens		Transfer		MetaUMT		MetaGUMT	
Parallel	Monolingual	De-En	En-De	De-En	De-En	En-De	De-En
5k	5k	28.04	32.74	30.31	34.31	**31.21**	**35.90**
8k	8k	28.86	33.14	30.51	35.22	**31.78**	**36.25**
16k	16k	29.62	33.88	31.49	36.62	**32.51**	**37.05**
32k	32k	30.55	35.35	33.25	37.25	**34.60**	**38.58**

Table T.3: Results of semi-supervised machine translation in the finetuning stage. "# tokens" indicates the number of tokens for both monolingual and parallel datasets. Each model is first pretrained on Medical, Law, EUbookshop, Koran, IT, and GlobalVoices and then finetuned to the low-resource domain (i.e., Law).

# tokens		Transfer		Mixed		MetaUMT		MetaGUMT	
En	De	De-En	En-De	En-De	De-En	En-De	De-En	En-De	De-En
5k	10k	25.84	31.67	26.04	31.90	28.80	32.65	**29.43**	**34.28**
8k	16k	25.95	31.83	26.09	32.01	27.84	32.93	**29.62**	**34.39**
16k	32k	25.96	32.03	26.44	32.37	27.92	32.96	**30.10**	**34.44**
32k	64k	27.34	32.64	27.39	32.84	28.67	33.52	**29.83**	**34.77**

Table T.4: Results on unbalanced monolingual data. This is the same results of Table 4 but included the additional baseline model, *Transfer*.