Neural Computing and Applications (2023) 35:14047–14060
https://doi.org/10.1007/s00521-023-08377-4

ORIGINAL ARTICLE

Effective domain awareness and adaptation approach via mask substructure for multi-domain neural machine translation

Shuanghong Huang[1,2] · Junjun Guo[1,2] · Zhengtao Yu[1,2] · Yonghua Wen[1,2]

Received: 8 December 2021 / Accepted: 13 February 2023 / Published online: 21 March 2023

Abstract
Multi-domain adaptation of neural machine translation (NMT) aims to learn a unified seq2seq framework based on multi-domain data. Domain corpus data mixing is one of the most important ways for multi-domain NMT, which has been widely explored in many recent works. However, due to the limitation of data mixing strategy, it often suffers from catastrophic forgetting problem or domain shift problem. To this end, we propose a domain-aware NMT with mask substructure. The mask substructure is employed in both Transformer-encoder and Transformer-decoder to capture domain-specific representations for each domain, then a domain fusion strategy is adopted to obtain a multi-domain adaptive NMT model. Our domain fusion framework could share domain-invariant knowledge and maintain domain-specific knowledge. We conduct extensive experiments on multi-domain NMT dataset, and the experimental results show significant improvements over the state-of-the-art (SOTA) approaches by up to 1.1 BLEU points on 8 domains and up to 4.5 BLEU points on an unseen domain. Moreover, the in-depth analysis shows that our model can also effectively alleviate both catastrophic forgetting and domain shift problems.

Keywords Neural machine translation · Multi-domain adaptation · Domain-aware mask substructure · Domain knowledge

1 Introduction

Neural machine translation (NMT) [1–3] is a seq2seq method for translating sentences from the source language to the target language, and (multi-)domain adaptation NMT is one of the crucial research directions which aims to construct a unified framework to translate multiple domain sentences. In domain NMT, due to the distinct style or domain terminology, there are many domain gaps for multi-domain data. Most previous works generally represent domain data in domain-specific semantic space with the domain elements, such as *Genre, Topic and Provenance* [4, 5], etc. However, due to domain style or vocabulary differences, traditional NMT approaches often suffer a performance drop. There is also substantial domain-invariant knowledge contained in each domain data. Therefore, it is possible to improve the machine translation performance with domain adaptation. How to build a unified encoder-decoder framework with domain-invariant knowledge to improve multiple domain translation performances is one of the critical issues for domain adaptation NMT.

The ultimate goal of (multi-)domain adaptation is to translate not only all domain corpus but also have a good performance within a unified model, and most domain translation tasks need to have high-quality and specific domain data. Unfortunately, two main issues are raised in domain NMT, (1) the large-scale high-quality parallel sentence pairs are scarce, and (2) the domain-specific terminologies are generally challenging to translate correctly. If we train the (multi-)domain NMT model in a traditional

✉ Junjun Guo
junjunguo_liip@kust.edu.cn

Shuanghong Huang
shuanghong@stu.kust.edu.cn

Zhengtao Yu
zhengtaoyu@kust.edu.cn

Yonghua Wen
wenyonghua@ymu.edu.cn

[1] Yunnan Key Laboratory of Artificial Intelligence, 650500 Kunming, China

[2] Faculty of Information Engineering and Automation, Kunming University of Science and Technology, 650500 Kunming, China

seq2seq framework, the model performance is usually unsatisfactory. Therefore, it is challenging to find a new domain adaptation method to obtain a unified and excellent model with all domain data.

As shown in Table 1, there are many differences between the *Spoken* domain and the *News* domain, e.g., the word 'apple'. It will inevitably suffer from the domain shift problem by only combing the domain-specific vocabularies and parallel sentence pairs together. However, in (multi-)domain NMT, the model obtained by simply mixing the training data of each domain cannot sufficiently reflect its domain characteristics because the domain features are lost during training. Due to the lack of representation and correspondence between words, the combination of domain terminology that can express its domain features is often not well translated.

To deal with the domain shift problem, there are two main domain adaptation strategies for domain NMT: (1) domain transfer strategy, using the corpus of the rich-resource domain (out-domain) to benefit and train the low-resource (in-domain) NMT model, and (2) domain adaptive strategy, using the multi-domain data to train a unified NMT model for all domain data. Domain transfer strategy focuses on a specific domain, while domain adaptive strategy mainly focuses on the generalization of multiple domains. We believe that building a unified multi-domain NMT model is conducive to sharing common knowledge between domains. Based on this, we mainly focus on domain adaptive strategy in this paper. The classical methods and current state-of-the-art (SOTA) models include fine-tuning [6], mixed fine-tuning [7], mixed with domain tags [8], domain-specific adapters [9], pruning then expanding [10], sequential prune-tune [11] and so on.

From our perspective, there is often semantic overlap between the training corpora data, leading to the domain data imbalance, and Fig. 1 illustrates the semantic overlap relationship between domains. These imbalanced data jointly training obtained model will suffer performance degradation in specific domains, affecting the final model performance. We argue that the parameter interference between domains ultimately causes the domain shift problem. Therefore, if we share most of the common domain-invariant knowledge and learn a small part of the private domain-specific knowledge according to the features of their respective domains during the training process, more specifically, the word-to-word representation and correspondence can be efficiently implemented based on their domain features, then we can obtain a unified multi-domain NMT model with well-performing through this method.

Table 1 A difference phenomenon caused by the combination of words exists in domain NMT

Spoken domain
ZH 他是个坏蛋，你最好离他远一点。
EN *He* is a *bad apple*, you'd better stay away from *him*.
News domain
ZH 苹果新闻是由苹果公司开发的新闻聚合应用程序。
EN *Apple News* is a news aggregator *app* developed by *Apple Inc*.

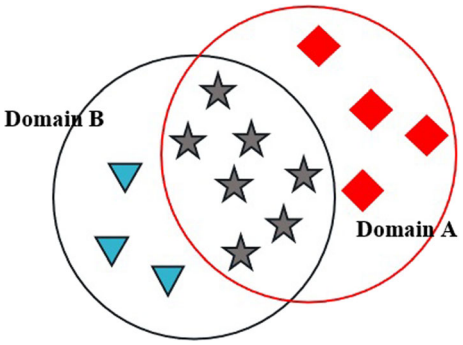


Fig. 1 Domain A and Domain B overlap in semantic space. The Five-pointed star represents common invariant knowledge between domains, the Triangle and the Square represent the private specific knowledge with respective domains

In addition, traditional domain NMT approaches also suffer from the catastrophic forgetting [12–14] problem. It is well known that if we jointly train all domain corpus will obtain an 'average' model, which will deviate from all domain models, and all training domains will be hurt because a multi-domain NMT model must allocate its model capabilities to match all domains. But suppose we use the 'average' model to fine-tune on respective domains, which will correct the deviation and obtain each domain-specific model with good performance, the description process is shown in Fig. 2.

To address the domain shift and the catastrophic forgetting problems, we first classified the multi-domain NMT model performance degradation as the domain shift problem caused by parameter interference between domains, and then we used the large-scale general domain data and a small amount of specific domain data to find a domain-

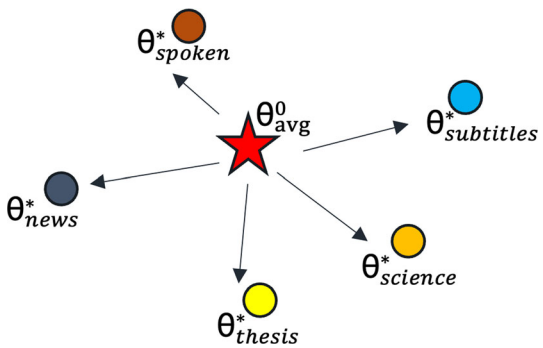


Fig. 2 Using the 'average' NMT model to fine-tune each domain, → represents the fine-tuning and correcting deviation process

aware mask substructure (DAMSS) according to the above analysis, which can effectively share the common domain-invariant knowledge and learn the private domain-specific knowledge, thereby the obtained model can effectively express the representation and correspondence with the combination of domain terminology words. By doing so, the final multi-domain NMT model can effectively generate respective domain results and represent respective domain semantic features in a unified model.

Our proposed method contributions are summarized as follows:

- This paper proposes a domain aware and adaptive NMT method based on DAMSS to alleviate catastrophic forgetting and domain shift problems.
- A simple and effective domain fusion strategy is adopted to obtain a multi-domain adaptive NMT model.
- A domain-aware adaptive Transformer is proposed with the DAMSS to extract domain-specific knowledge for each domain.
- Experimental results on the multi-domain dataset are given to show the effectiveness of our proposed domain awareness and adaptation NMT approach.

The rest of this paper is organized as follows. Section 2 discusses related work from two aspects: training methods and model structure. Our proposed methods will be presented in detail in Sect. 3, followed by experiments in Sect. 4. Experimental results and analysis are given in Sect. 5. Section 6 concludes our work.

2 Related work

As far as we know, our research work is obviously related to training methods and model structure.

2.1 Training methods

Fine-tuning [6, 15, 16] on the general domain (out-domain) model with in-domain data is the easiest way to improve, which will significantly damage the general domain translation quality. Dakwale and Monz [17] improved fine-tuning method through the knowledge distillation (KD) method [18], which can maintain good translation performance of out-domain. Chu et al. [7] proposed a mixed fine-tuning method, which fine-tunes the general domain model with the mix of the general domain data and over-sampling in-domain data. Barone et al. [19] added regularization terms to alleviate the over-fitting phenomenon of in-domain model during fine-tuning, Khayrallah et al. [20] and Thompson et al. [21] also added regularization terms to make the model parameters closer to their original values. Wang et al. [22] trained on all domain corpus but assigned different learning rate weights to different sentences, and the weights were set according to Axelrod et al. [23] proposed data screening indicators. Chen et al. [24] trained a domain classifier and set the data weights according to the domain classifier scores. Vilar [25] effectively assigned weight parameters using the out-domain model hidden state to adapt in-domain data and freeze out-domain parameters. Yan et al. [26] accurately set the weights to word level and set the learning weights for each word by the difference between the scores of the in and out domain language models. Recently, Zhang et al. [27] applied curriculum learning to domain adaptation and the model can gradually transition from out-domain to in-domain through the curriculum design. Zeng et al. [28] proposed an iterative training method, which allows the out-domain and in-domain models to iteratively learn each other knowledge based on knowledge distillation method. Wang et al. [29] applied efficient lifelong learning to domain adaptation through establishing complementary learning systems. Gu et al. [10] fixed the important parameters and pruned the unimportant parameters, and Liang et al. [11] also used a similar idea, the difference is whether adopt knowledge distillation method to maintain general domain performance.

2.2 Model structure

In addition to the deep fusion model proposed by Gulcehre et al. [30] and Dou et al. [31], Nguyen and Chiang [32] augmented models by using a lexical choice network. Britz et al. [33] and Wang et al. [34] mixed data from different domains for training and introduced a discriminator to extract common features from respective domains at the same time. Kobus et al. [35] added domain-specific tags to each sentence, so that the model can distinguish domain of the input data, this approach can be extended to new domains by adding more labels [8], or by defining multi-dimensional domain tags [36]. Thompson et al. [37] and Wuebker et al. [38] pointed out that most of the out-domain model parameters can be fixed, and only a small part of the in-domain parameters need to be fine-tuned. Gu et al. [39] preserved the domain-specific features by adding domain-specific modules to the model. Wu et al. [40] added an explicit multi-dimensional domain embedding based on Gu et al. [39], Zeng et al. [41] proposed a method with word-level domain context discrimination to determine domain-specific and domain-shared source sentence representations. Su et al. [42] improved the multi-domain NMT model by using multi-task learning and monolingual attention-based domain classification tasks. Jiang et al. [43] defined a domain-specific attention network that can be activated at

the word level instead of the sentence level. Bapna et al. [9] injected domain-specific adapter modules into each layer of the general domain model and fine-tuned them by freezing general domain parameters. The adapter layers are typically inserted between the encoders and decoders, and Pham et al. [44] used a domain discriminator to determine which adapter to use. Important decisions like adapter size and the number of training steps can be tuned manually or determined via a meta-learning method when using adapters for NMT domain adaptation as in Sharaf et al. [45].

As stated above, significantly different from the most of above methods, along with the studies of fine-tuning [6, 16, 46], sparse sharing for multiple tasks [47], and LaSS for multilingual NMT [48], we use the large-scale general domain data and a limited number of specific domain data to obtain a unified multi-domain NMT model based on the DAMSS, which can effectively share the common domain-invariant knowledge and learn private domain-specific knowledge. To the best of our knowledge, our work is the first attempt to explore such a DAMSS for multi-domain adaptation NMT.

3 Method

This section will describe the framework and training strategy in detail. We adopt the Transformer [3] as the backbone network, which has the encoder-decoder architecture, and the whole architecture of our research is illustrated in Fig. 3. In addition to the basic structure of Transformer, it contains word embedding with domain tags, and both Transformer-encoder and Transformer-decoder apply DAMSS, which consists of the mask-based multi-domain attention mechanism.

Our goal is to use the large-scale general domain data and a small amount of specific domain data based on DAMSS to obtain a unified multi-domain NMT model with good performance, which can solve the catastrophic forgetting and domain shift problems, thereby improving the translation performance of multiple domains. Through the proposal of DAMSS, the model network parameters can be automatically adjusted according to the domain and enhance the entire model's adaptability.

In this paper, we formulate the multi-domain NMT task as follows. Denote D_g and $\{D_s \mid s = 1, 2, \cdots\}$ as the large-scale general data and the specific domain data, respectively, where s is the index of specific domain. Let x be a source sentence consisting of words $\{x_1, x_2, \cdots, x_i\}$, y denotes the ground-truth sentence consisting of words $\{y_1, y_2, \cdots, y_j\}$, the domain tags $\{d_k \mid k = 0, 1, \cdots\}$ correspond to the respective sentence pair (x, y), where k the index of domain tag, and $\hat{\mathrm{y}}$ be a translation sentence consisting of words $\{\hat{y}_1, \hat{y}_2, \cdots, \hat{y}_{j^*}\}$. The scoring function $f(\mathrm{y}, \hat{\mathrm{y}}) \in \mathbb{R}$ is employed to calculate BLEU.

3.1 Domain word embedding

Different from the previous traditional word embedding, which just uses the BOS (begin of sentence), PAD (padding word), UNK (unknown word) and EOS (end of sentence) tokens in a sentence. Inspired by Kobus et al. [35], we apply the domain tags to convey the domain of a sentence pair. In our framework, as shown in Fig. 4, if given the sentence pair (x, y) and its domain tag d_k, we insert the DTS (Domain Tags) token which consists of $\{d_k \mid k = 0, 1, \cdots\}$ to indicate its domain at the beginning of the sentence, so that a sentence word embedding from source or target consists of domain tag and sentence, which can preserve the greatest extent domain information.

More specifically, domain information will be incorporated into embedding via domain tags, which can ameliorate traditional word embedding. To better describe our proposed domain word embedding method, we summarize the method function in Eq. (1), the source sentence x and the target sentence y which come from the k-th domain tag are embedded as follows:

$$\begin{aligned} \mathrm{E}_s &= \mathrm{emb}_s(d_k \oplus \mathrm{x}) \\ \mathrm{E}_t &= \mathrm{emb}_t(d_k \oplus \mathrm{y}) \end{aligned} \tag{1}$$

where E_s and E_t denote the results of word embedding process for the source and target, respectively, emb_s and emb_t represent the word embedding process and the '$\oplus$' represents the connect operation.

In addition, similar to most domain adaptation NMT approaches [7, 27], the domain sentences in respective domains corpus are shuffled, but the order of domain corpus used to train the multi-domain NMT model will not be changed in the training process.

3.2 Domain-aware mask subStructure

In Fig. 3, the left column shows the traditional Transformer Encoder component, which has the multi-head self-attention mechanism, the fully connected feed-forward network, and employed residual connections around each of the sub-layer, followed by layer normalization. In addition to these features, as shown in an enlarged portion of the dashed line, we have implemented DAMSS, which consists of the mask-based multi-domain attention mechanism.

More specifically, the mask-based multi-domain attention mechanism contains the domain-aware adaptive mask with respective domain knowledge, and the red box in

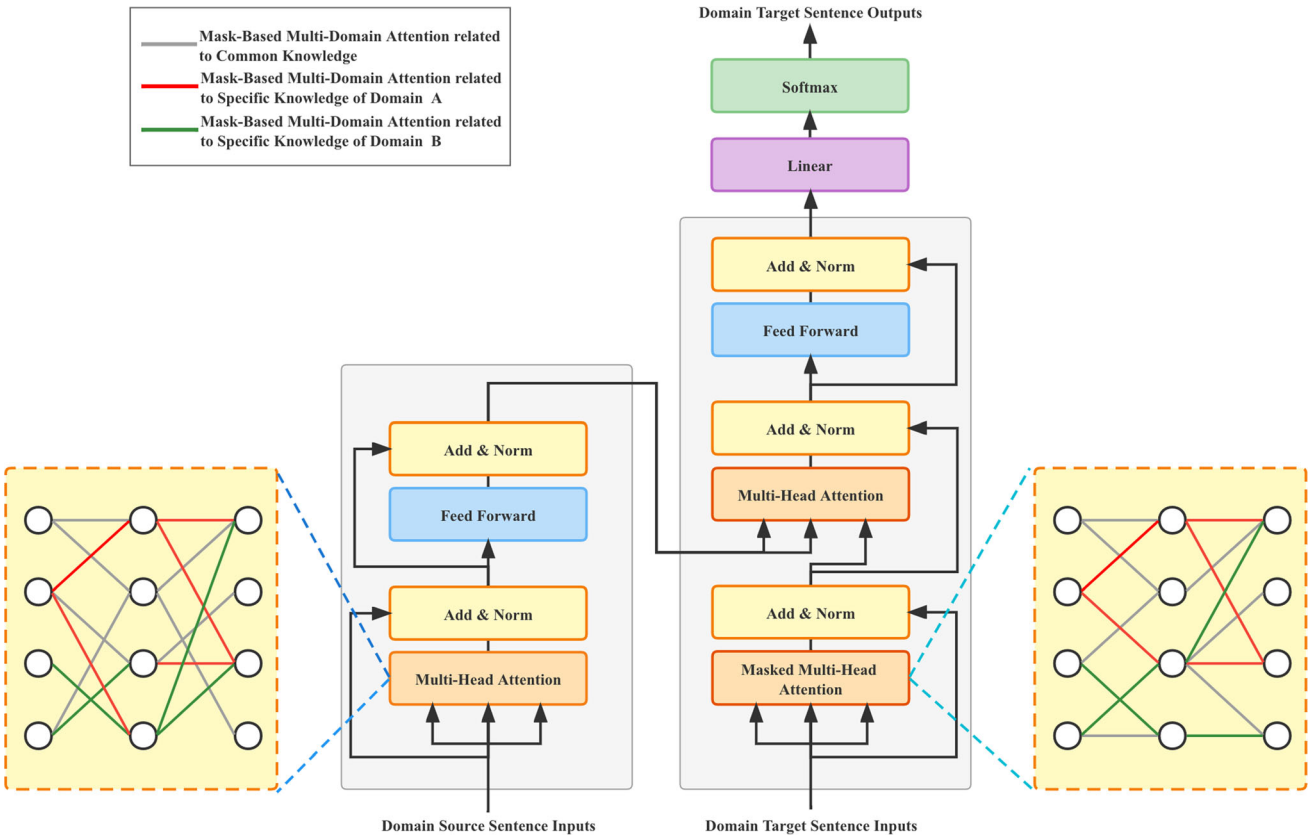


Fig. 3 Illustration of the whole architecture. The model is made up of Encoder (left column) and Decoder (right column) subnetworks

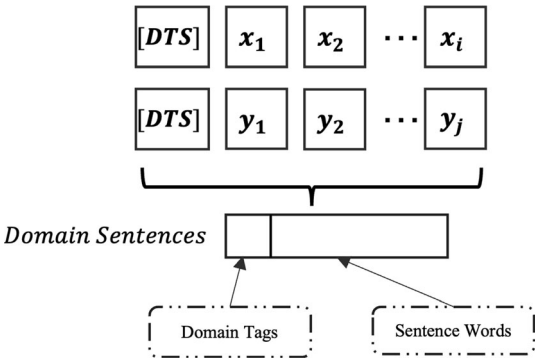


Fig. 4 Illustration of domain sentences composition process, the '[DTS]' denotes the domain tags $\{d_k \mid k = 0, 1, \cdots\}$ correspond to the respective sentence pair (x, y)

Fig. 5 indicates the detailed composition. Given a 6-layer Transformer model with the parameters $P_{Trans} = \{P_i \mid i = 1, \cdots, 6\}$, and the encoder parameters are $P_i^{Enc} = \{P_{attn}^{Enc}, P_{FFN}\}$ where the P_{attn}^{Enc} and P_{FFN} represent the multi-head self-attention and feed-forward network module parameters, respectively, and P_{attn}^{Enc} consists of concrete values $\{W_k \mid k = 0, 1, \cdots\}$. We modified the Encoder Attention function of Transformer to Eq. (2) to better describe the change.

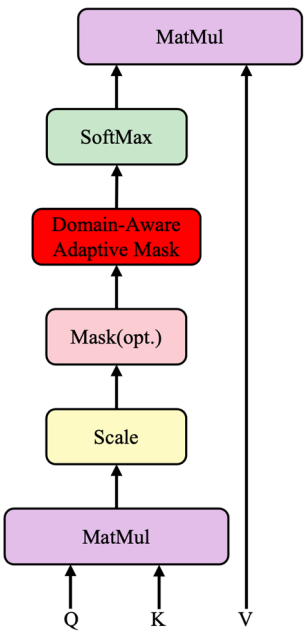


Fig. 5 Scaled dot-product attention with domain-aware adaptive mask, which contains the respective domain knowledge

$$\text{Attention}_{Enc}(Q,K,V) = \text{softmax}\left(f\left(\frac{QK^T}{\sqrt{d_k}}, \mathcal{M}^{Enc}\right)\right)V \tag{2}$$

and

$$\mathcal{M}^{Enc} = \begin{cases} 0, & |W_k| < \alpha \\ 1, & |W_k| \geq \alpha \end{cases} \tag{3}$$

where f represents processing $\frac{QK^T}{\sqrt{d_k}}$ with mask $\mathcal{M}^{Enc}$, and the α represents a hyperparameter. Besides, the value in the domain-aware adaptive mask $\mathcal{M}^{Enc}$ is 1 represents the parameter weight is reserved, and 0 will abandon the parameter weight.

The DAMSS in Encoder can efficiently capture relevant knowledge according to their respective domains and prevent model translation performance degradation due to the loss of their respective domain features during the encoding process. As far as we know, the sequential prune-tune method proposed by Liang et al. [11] also utilizes the mask matrix to improve the model performance. The mask-based methods can dynamically train the model network parameters based on their respective domain knowledge, significantly improving the ability to share the common domain-invariant knowledge between domains and learn the private domain-specific knowledge separately.

Similar to the Encoder, apart from the typical components in Transformer, the DAMSS is also applied in the Decoder, as shown in an enlarged portion of the dashed line in the right of Fig. 3, and the detailed composition is shown in Fig. 5.

To be more specific, the decoder parameters are $P_i^{Dec} = \{P_{attn}^{Dec}, P_{FFN}\}$, where the P_{attn}^{Dec} represents the multi-head self-attention and multi-head cross-attention module parameters, the P_{FFN} represents the feed-forward network module parameters, and P_{attn}^{Dec} consists of concrete values $\{W_k \mid k = 0, 1, \cdots\}$, we also modified the Decoder Attention function to Eq. (4).

$$\text{Attention}_{Dec}(Q,K,V) = \text{softmax}\left(f\left(\frac{QK^T}{\sqrt{d_k}}, \mathcal{M}^{Dec}\right)\right)V \tag{4}$$

and

$$\mathcal{M}^{Dec} = \begin{cases} 0, & |W_k| < \alpha \\ 1, & |W_k| \geq \alpha \end{cases} \tag{5}$$

where f represents processing $\frac{QK^T}{\sqrt{d_k}}$ with mask $\mathcal{M}^{Dec}$, and the α represents a hyperparameter.

The DAMSS in Decoder allows the model to jointly attend to information from different sentences representation in respective domains and map the relationship between words combined with the characteristics of respective domains according to the given sentence. Furthermore, the DAMSS in Decoder has similar functions to Encoder. It can effectively adapt to the decoding according to respective domains so that the model can capture domain-related information to improve performance.

3.3 Training strategy

To better describe our framework, we summarize the training loss function in Eq. (6) and the training procedure in Algorithm 1.

$$\mathcal{L}(\theta) = \sum_{(\mathrm{x},\mathrm{y})\in d_k} -\log P(\mathrm{y} \mid \mathrm{x}; \theta) \tag{6}$$

where (x, y) presents a sentence pair from respective domain tags d_k and the θ denotes the NMT model parameters.

Specifically, we first train a base multi-domain NMT model θ^0 based on the large-scale general domain data and all domain-specific data, and we fine-tune the base multi-domain NMT model to obtain each specific domain model θ_s^*, then we utilize the average-pool mechanism to get each domain-specific mask M_s based on the attention matrixes of all attention modules in θ_s^*, the domain-aware adaptive mask $\mathcal{M}$ consists of all domain-specific mask matrixes, which can be shown as follows:

$$\mathcal{M} = \{M_s \mid s = 1, 2, \cdots\} \tag{7}$$

where s is the index of specific domain.

The unified multi-domain modal θ^* is fine-tuned based on the base multi-domain model θ^0 with the aid of the domain-aware adaptive mask $\mathcal{M}$. During this process, the multi-domain NMT model can represent respective domain parameters that contain the special domain knowledge and share the common and invariant parameters between domains in the meanwhile. We try to find such DAMSS that can represent respective domains and improve model generalization ability. Figure 6 illustrates our training strategy.

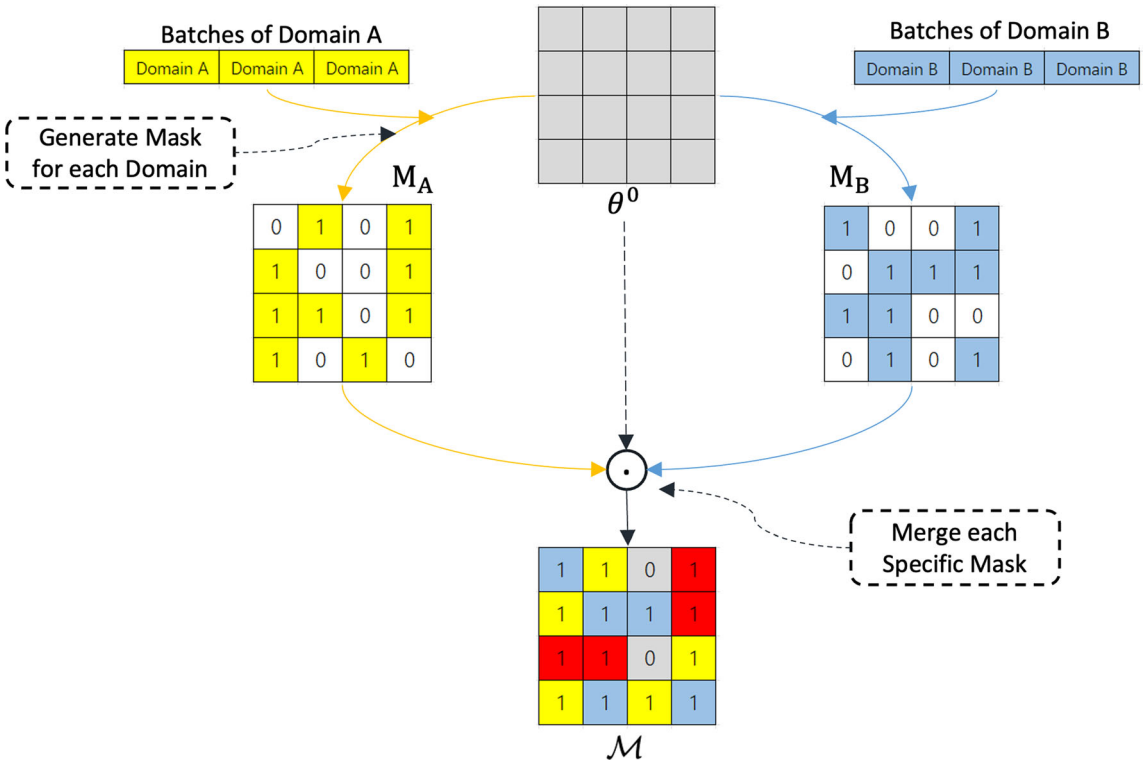


Fig. 6 Gray squares are the base multi-domain model θ^0 parameters. Orange and blue squares indicate retaining the parameter weights, while white squares represent the parameter weights needed to abandon. Red squares represent both parameter weights required to retain for domain A and B. $\odot$ means merging each domain mask process

Algorithm 1 Effective Domain Awareness and Adaptation Approach via Mask Substructure for Multi-Domain NMT

Input: Training sets $\{D_g,\ D_s\}$, development sets $\{D_g^v,\ D_s^v\}$, where s is the index of specific domain.
Output: Multi-Domain NMT model θ^*.
$\theta^0 \leftarrow \text{TrainBaseModel}(\{D_g, D_s\},\ \{D_g^v, D_s^v\})$
for $s = 1, 2, \cdots$ **do**
  $\theta_s^* \leftarrow \text{TrainSpecificModel}(\theta^0,\ D_s,\ D_s^v)$
  if (Attention Module) in θ_s^* :
    Generate M_s
end for
$\mathcal{M} = \{M_s \mid s = 1, 2, \cdots\}$
$\theta^* \leftarrow \text{TrainMulti-domainModel}(\{D_g, D_s\},\ \{D_g^v, D_s^v\},\ \mathcal{M})$

4 Experimental setup

4.1 Dataset

In our experiments, we take the WMT[1] corpus as the large-scale general domain data and use the UM-Corpus[2] [49] as the limited number of specific domain data. For WMT, we collect about 5.5 M Chinese-English general domain language pairs. For UM-Corpus, which is categorized into eight different text domains including *Education*, *Laws*, *Microblog*, *News*, *Science*, *Spoken*, *Subtitles* and *Thesis*, we exclude the *Microblog* domain data from UM-Corpus and keep the rest as the specific domains' experimental data. Besides, the *newsdev2017* and *newstest2017* are chosen as the development and test set for the general domain, respectively. We filter out the duplicate sentences for

[1] https://www.statmt.org/.
[2] http://nlp2ct.cis.umac.mo/um-corpus/.

Table 2 The detailed description of each domain dataset

Domain	Training	Development	Test
General	5.5 M	2K	2K
Education	200K	1.5K	1.5K
Laws	400K	1.5K	1.5K
News	400K	1.5K	1.5K
Science	200K	1.5K	1.5K
Spoken	200K	1.5K	1.5K
Subtitles	200K	1.5K	1.5K
Thesis	200K	1.5K	1.5K

specific domains and then randomly extract a certain proportion of the corpus as the training, development, and test set for respective domains. The detailed description of each domain dataset is shown in Table 2.

4.2 Data preprocessing and evaluation metrics

We first employ *Stanford Segmenter*[3] to execute word segmentation on Chinese sentences and *MOSES script*[4] to tokenize and truecase the English sentences. And then, we limit the proportion of sentences to 1.5 and the length of sentences to 200 words for respective domains. Besides, we apply *Byte Pair Encoding* (BPE) [50] to split words into sub-words and set the vocabulary size for Chinese as 40,000 and English as 30,000. Finally, we evaluate the translation quality with BLEU scores [51] as calculated by *multi-bleu.perl* script.

4.3 Model settings

Our experimental environment is Ubuntu 20.04 based on the Linux system, the compiled language and version are python 3.7.0. We use *Facebook AI Research Sequence-to-Sequence Toolkit*[5] (Fairseq) [52] version 0.10.2 based on PyTorch as the sequence modeling toolkit, which can train custom models for translation.

In this paper, we first employ experiments on Transformer-Base [3], but the experimental results in the general domain are not well due to a large amount of training data. Based on this, we replace the architecture with Transformer-Big. As shown in Table 3, there is a considerable difference between the general domain results in the Transformer-Base and Transformer-Big row, and we think that the training parameters of Transformer-Base are not enough to support the entire model when the training data scale reaches a certain level, and the influence not very significant for other domains where the training data is not very massive.

We choose Transformer-Big as our backbone network and follow Vaswani et al. [3] to set the configurations. First, the dimension of all input and output layers is 1024, the feed-forward network layer is 4096, and we employ 16 parallel attention heads in both encoder and decoder. Then, parameter optimization is performed using Adaptive moment estimation (*Adam*) [53], and the learning rate will dynamically adjust during the training process. Besides, We batch sentence pairs by approximated length and limit input and output tokens per batch to 4096 tokens, the hyperparameter α in DAMSS is 0.7. Finally, we employ oversampling for specific domains to balance the training data distribution with a temperature of $T = 3$ and also use dropout = 0.3 to prevent the over-fitting effectively. As for decoding, we employ beam search algorithm and set the beam size as 5. For simplicity, we only report the best BLEU from the best multi-domain NMT model.

4.4 Compared methods

To verify the effectiveness of our framework, we compare with some classical methods and the current state-of-the-art (SOTA) methods, namely:

- *Transformer-Big* [3]. A single specific domain NMT model trained on their respective domain corpus with Transformer-Big.
- *Fine-tuning (FT)* [6]. It first trains on large-scale general domain corpus and then fine-tunes it using a limited number of specific domain corpus to continue training.
- *Mixed Fine-tuning (MFT)* [7]. It first trains on large-scale general domain corpus and then fine-tunes it using both large-scale general domain corpus and oversampling a limited number of specific domain data.
- *Mixed with Domain Tags (MDT)* [8]. A multi-domain NMT model trained on the mix of large-scale general domain and a small amount of specific domain corpus but added the domain tags before the corpus of respective domains.
- *Pruning Then Expanding (PTE)* [10]. A model trained on general domain and pruned, then use specific domains data and knowledge distillation where the unpruned model as a teacher and the pruned model as a student to adjust the pruned model, finally expand and fine-tune the pruned model to the original size.
- *Sequential Prune-Tune (SPT)* [11]: Finding and freezing the most informative parameters on the general domain model, then pruning unnecessary parameters, finally

[3] https://nlp.stanford.edu/.

[4] http://www.statmt.org/moses/.

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

Table 3 Results on WMT (General Domain) dataset and UM-Corpus (Specific Domains) dataset

Model	Domains								Avg.
	General	Education	Laws	News	Science	Spoken	Subtitles	Thesis	
Transformer-Base [3]	15.52	15.35	33.31	18.24	22.70	20.46	30.53	18.03	21.77
Transformer-Big [3]	23.70	15.46	34.90	18.41	23.80	20.29	**32.40**	18.15	23.39
FT [6]	19.50	16.91	31.56	18.99	25.74	30.26	22.78	14.89	22.58
MFT [7]	21.55	18.65	32.85	19.97	32.71	32.71	30.05	20.04	26.07
MDT(Baseline) [8]	23.20	19.92	36.15	19.49	30.68	**35.84**	25.44	19.37	26.26
PTE [10]	23.78	19.55	–	–	–	18.69	–	16.85	19.72
SPT [11]	–	**31.20**	**50.30**	**21.30**	–	14.60	17.20	16.20	25.13
Ours	**23.85**	21.45	36.55	**21.29**	**33.21**	34.89	27.51	**20.15**	**27.36**

The bold font in the table is to highlight the best-performing value in this column

All models are based on Transformer-Big. Ours consistently outperforms the Baseline model in most domains and keeps translation performance in the general domain

using mask matrix and specific domain data to fine-tune the domain-specific sub-networks.

5 Results and analysis

5.1 Main results

Table 3 shows the main experimental results. For *PTE* and *SPT*, since they are recent works about domain adaptation, and the training data is consistent with ours, which was sampled from the WMT and UM-Corpus, we directly quoted the results. For specific domains, our framework significantly outperforms the Baseline model. Furthermore, we reach the following conclusions:

First, we calculate the average of BLEU to better compared with the recent research works *PTE* and *SPT*, our method averaged more than 7.64 and 2.23 compared to *PTE* and *SPT*, respectively. Besides, the *SPT* also used the strategy of mask-based to retain domain-specific information, so comparing the results can further verify the effectiveness and reflect the model generalization ability of our proposed method.

Second, for most specific domains, our proposed method surpasses *Transformer-Big, FT, MFT and MDT (Baseline)*, commonly used in the domain adaptation NMT. This confirms the effectiveness of using the DAMSS to obtain a unified multi-domain NMT model with good performance through large-scale general domain data and a small amount of specific domain data. We can confirm that our method can alleviate the domain shift problem caused by parameter interference between domains, thereby improving the multi-domain NMT model translation performance.

Finally, compared with *FT* and *MFT*, our model can effectively avoid the problem of catastrophic forgetting in the general domain. Although *MDT (Baseline)* and *PTE* can also solve this problem to a certain extent, the effect of our proposed framework is more pronounced. The underlying reason is that our multi-domain NMT model based on DAMSS can discriminate domain-specific and domain-shared information to alleviate the performance degradation problem in general domain and improve specific domain's performance.

5.2 Ablation experiment

5.2.1 Effects of DAMSS component

In this section, for simplicity, we explored DAMSS components' effects using the General domain and another specific domain (Science, which with the highest performance improvement). The results are shown in Table 4. First, comparing *DAMSS_Enc* and *DAMSS_All*, we can confirm the effectiveness of *DAMSS_Dec*, which can capture domain-related information according to respective domains during the decoding process. Second, comparing *DAMSS_Dec* and *DAMSS_All*, we can also confirm the effectiveness of *DAMSS_Enc*, which can extract and learn the domain-specific knowledge during the encoding process. Finally, comparing all components simultaneously, it is shown that combining *DAMSS_Enc* and *DAMSS_Decc* can further improve the model performance, and both are indispensable.

Besides, note that the model performance will be further improved when both *DAMSS_Enc* and *DAMSS_Dec* exist simultaneously. This result demonstrates the advantage of DAMSS under our framework. Moreover, all components of DAMSS will influence each other to share the common domain-invariant knowledge and maintain the respective private domain-specific knowledge.

5.2.2 Comparison with different mask type

To verify whether mask type influences our proposed DAMSS which consists of the mask-based multi-domain attention mechanism, we use a *Random Mask* and a general domain mask (*Single Mask*) to make a comparison, and the result is shown in Table 5. We know that the multi-domain attention mechanism based on the domain-aware adaptive mask has the best effect compared to other mask types, so we can determine that the domain-aware adaptive mask is effective.

The domain-aware adaptive mask can avoid parameter interference between domains according to the extracted and learned domain-specific knowledge, thereby achieving the domain adaptation process to solve the domain shift problem better, which is conducive to improving the model performance.

5.2.3 Model generalization ability

As shown in Table 6, we used the *DAMSS_All* from Sect. 5.2.1 to translate other domains and compared the results with the DAMSS from main experiment 5.1. Compared with *DAMSS_All*, although the BLEU score in *General* and *Science* domain has decreased, the decline is not particularly dramatic. Besides, we tested the *DAMSS_All* to other domains, and from Table 6, it is shown that there is a dramatic decline in *Laws* and *Subtitles* domain. This result strongly proves the generalization ability of our proposed method, which can keep the performance of the general domain and improve the performance of specific domains.

5.3 Other experiments

5.3.1 Domain similarity and model representation ability

Ideally, similar domains should share more parameters because they have more overlapping domain features. Therefore, we use the language model of respective domains to represent their domain, then calculate the distance between each domain to represent the domain similarity relationship and the result is shown in Fig. 7a.

Table 4 Results of comparison with DAMSS component

DAMSS Component	General	Science
DAMSS_Enc	23.84	33.27
DAMSS_Dec	23.53	33.71
DAMSS_All	**24.00**	**33.78**

The bold font in the table is to highlight the best-performing value in this column

Table 5 Results of comparison with different mask types

Mask type	General	Science
Random mask	22.79	28.00
Single mask	23.62	32.96
Domain-aware adaptive mask	**24.00**	**33.78**

The bold font in the table is to highlight the best-performing value in this column

Furthermore, we first randomly select a sentence from the Test sets in respective domains, then use our model to characterize the sentence, and finally measure the distance of two domain sentences by *Hellinger distance*, which is defined as Eq. (8) when the probability distribution is discrete, where P and Q are the discrete distributions for the true target vector and domain representation vector, V is the vocabulary size, and the result is shown in Fig. 7b.

$$H(P,Q) = \frac{1}{\sqrt{2}}\sqrt{\sum_{i=1}^{V}(\sqrt{p_i} - \sqrt{q_i})^2} \tag{8}$$

From Fig. 7, we can see that model similarity is positively correlated with the model representation ability. And the most obvious improvement effect is also concentrated on the dark blue domain, we guess that more invariant domain knowledge is shared between domains, and the specific domain knowledge can be extracted from respective domains during the training process, thereby solving the domain shift problem caused by parameter interference between domains, which can achieve the process of domain adaptation and make the model translation performance better.

5.3.2 Extensibility to new domain

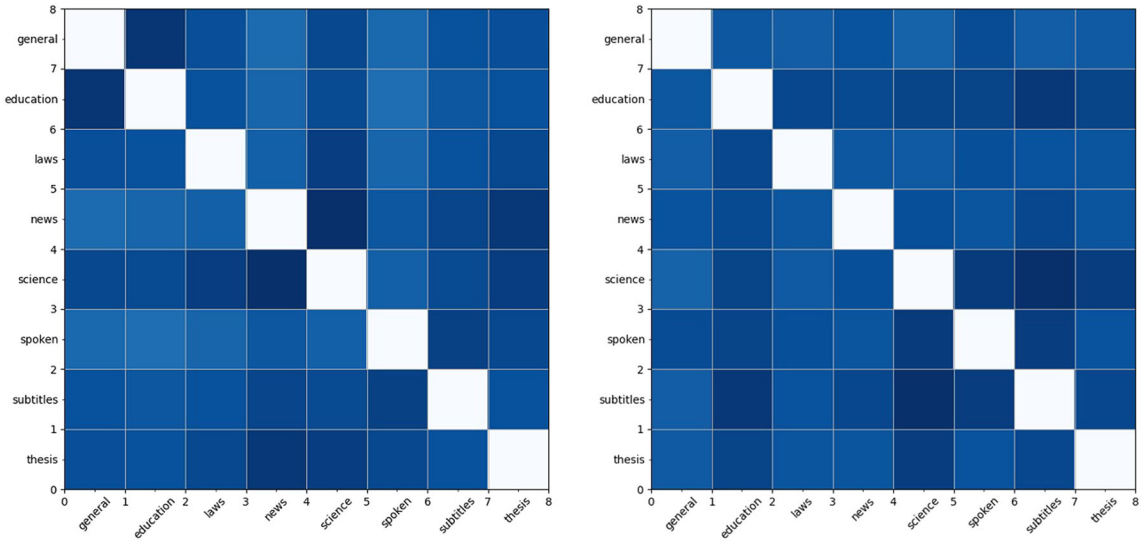
We show that our method can quickly adapt to new unseen domains without a dramatic drop for other existing domains. Specifically, we used the Test sets from United Nations Parallel Corpus[6] as a new domain called the *Conference* domain. In the generation process, we treated it as *General* domain to side-verify the model generalization capability. As shown in Table 7, contrasted with the *Transformer-Big*, there are varying degrees of decline in the *FT*, *MFT*, *MDT (Baseline)* models. In contrast, our model did not decline and achieved the highest result.

Compared with previous works, we verify the extensibility of DAMSS on *Conference* domain, showing that our proposed method can quickly and effectively extend to new domains and have a good performance, which has important practical significance for rapid adaptation to save time. Furthermore, we attribute the easy adaptation for new

[6] https://conferences.unite.un.org/UNCorpus.

Table 6 Results of overall model generalization ability

	General	Education	Laws	News	Science	Spoken	Subtitles	Thesis
DAMSS_All	**24.00**	**22.33**	30.02	20.33	**33.78**	**35.62**	20.61	19.42
Main Experiment	23.85	21.45	**36.55**	**21.29**	33.21	34.89	**27.51**	**20.15**



(a) Domain Similarity (b) Model Representation Ability

Fig. 7 Heat map of model similarity and model representation ability between domains. The model similarity is positively correlated with the model representation ability

Table 7 Test results of respective model in the *Conference* domain

Model	Conference
Transformer-Big	36.59
FT	28.27
MFT	28.84
MDT(Baseline)	34.57
Ours	**39.10**

The bold font in the table is to highlight the best-performing value in this column

Table 8 Results of respective model parameters size

Model	Parameters size
Transformer-Big	96.5 M
FT	96.5 M
MFT	96.5 M
MDT(Baseline)	96.5 M
PTE	100.5 M
Ours	96.5 M

domains to the DAMSS, which can share common domain-invariant knowledge and extract domain-specific knowledge of respective domains, thereby can fast adaptation toward new domains.

5.3.3 Comparison with model parameters size

As shown in Table 8, compared to other models, our model parameters size is the same as most, but combined with the main experiment results, our method can achieve better results when the model parameters size is consistent, which further proves the effectiveness of our approach and the generalization ability of our model. We argue that the DAMSS can learn more domain-related information and knowledge in limited model parameters. In addition, it has a particular meaning for deploying models with multiple domains to save space in practical applications.

5.3.4 Example analysis

We sample a few translation examples from *Spoken* and *News* domain, as shown in Table 9, compared with *Baseline*

Table 9 Our model and Baseline model translation results

Spoken domain	
Src	他是个坏蛋，你最好离他远一点。
Ref	He is a bad apple, you'd better stay away from him.
Baseline	He's a bad guy, you'd better keep an eye off him.
Ours	He's a bad guy, you'd better *get away from* him.
News domain	
Src	苹果新闻是由苹果公司开发的新闻聚合应用程序。
Ref	Apple News is a news aggregator app developed by Apple Inc.
Baseline	Apple news is an app for news aggregation developed by Apple.
Ours	Apple news is a *news aggregation application* developed by Apple.

translation results, we observe that the *Baseline* has a severe 'reference is unclear' problem, and our method can significantly alleviate the issue.

This result can further verify the validity of our model, namely, our method can not only pay attention to the association between words at the sentence level but also resolve the parameter interference between domains by sharing common domain-invariant knowledge and learning domain-specific knowledge of respective domains, then solving the domain shift problem in the process of domain adaptation, finally improving specific domains translation performance and keeping the general domain performance.

6 Conclusion

In this paper, we have proposed an effective DAMSS framework for multi-domain adaptation NMT, which can alleviate the problem of catastrophic forgetting in *General* domain and improve the translation performance of specific domains. Furthermore, DAMSS can extend easily to new domains without a dramatic decline and keep the performance of existing domains. Finally, Experimental results and in-depth analyses on translation tasks strongly demonstrate the effectiveness of our research, which can alleviate both catastrophic forgetting and domain shift problems by sharing common domain-invariant knowledge and learning domain-specific knowledge according to respective domains. In future, we plan to extend our framework to low-resource multi-domain translation and apply our framework to other translation models.

Funding Funding was provided by National Natural Science Foundation of China (Grant No. 61732005, Grant No. 61866020, Grant No. U21B2027, Grant No. 61972186), Yunnan provincial major science and technology special plan projects (Grant No. 202002AD080001, Grant No. 202103AA080015).

Data availability All data generated or analyzed during this study are included in this published article.

Declarations

Conflict of interest The authors declared that they have no conflict of interest to this work. We declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with the work submitted.

References

1. Cho K, van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y (2014) Learning phrase representations using rnn encoder–decoder for statistical machine translation. In: Proceedings of the 2014 conference on empirical methods in natural language processing, 1724–1734
2. Bahdanau D, Cho K.H, Bengio Y (2015) Neural machine translation by jointly learning to align and translate. In: 3rd international conference on learning representations, ICLR 2015
3. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A. N, Kaiser Ł (2017) Polosukhin I.: Attention is all you need. In: Advances in neural information processing systems, 5998–6008
4. van der Wees M (2017) What's in a domain? towards fine-grained adaptation for machine translation
5. Saunders D (2022) Domain adaptation and multi-domain adaptation for neural machine translation: a survey. Artif Intell Res 75:351–424
6. Luong M.-T, Manning C.D, (2015) Stanford neural machine translation systems for spoken language domains. In: Proceedings of the international workshop on spoken language translation. Da Nang, Vietnam
7. Chu C, Dabre R, Kurohashi S (2017) An empirical comparison of domain adaptation methods for neural machine translation. In: Proceedings of the 55th annual meeting of the association for computational linguistics (Vol 2: Short Papers), pp 385–391
8. Tars S, Fishel M (2018) Multi-domain neural machine translation. In: Proceedings of the 21st annual conference of the European association for machine translation: 28-30 May 2018, Universitat d'Alacant, Alacant, Spain, pp 259–268 . European Association for Machine Translation
9. Bapna A, Firat O (2019) Simple, scalable adaptation for neural machine translation. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th

international joint conference on natural language processing, 1538–1548
10. Gu S, Feng Y, Xie W (2021) Pruning-then-expanding model for domain adaptation of neural machine translation. In: Proceedings of the 2021 conference of the North American chapter of the association for computational linguistics: human language technologies, 3942–3952
11. Liang J, Zhao C, Wang M, Qiu X, Li L (2021) Finding sparse structures for domain specific neural machine translation. Proc AAAI Conf Artif Intell 35:13333–13342
12. McCloskey M, Cohen NJ (1989) Catastrophic interference in connectionist networks: the sequential learning problem. Psychol Learn Motiv 24:109–165
13. Ratcliff R (1990) Connectionist models of recognition memory: constraints imposed by learning and forgetting functions. Psychol Rev 97(2):285
14. Chu C, Wang R (2018) A survey of domain adaptation for neural machine translation. In: Proceedings of the 27th international conference on computational linguistics, 1304–1319
15. Freitag M, Al-Onaizan Y (2016) Fast domain adaptation for neural machine translation. CoRR **abs/1612.06897**
16. Servan C, Crego J, Senellart J (2016) Domain specialization: a post-training domain adaptation for neural machine translation. CoRR **abs/1612.06141**
17. Dakwale P, Monz C (2017) Finetuning for neural machine translation with limited degradation across in-and out-of-domain data. In: Proceedings of the XVI machine translation summit, 117
18. Hinton G, Vinyals O, Dean J (2015) Distilling the knowledge in a neural network. stat 1050, 9
19. Barone A.M, Haddow B, Germann U, Sennrich R (2017) Regularization techniques for fine-tuning in neural machine translation. In: Proceedings of the 2017 conference on empirical methods in natural language processing, 1489–1494
20. Khayrallah H, Thompson B, Duh K, Koehn P (2018) Regularized training objective for continued training for domain adaptation in neural machine translation. In: Proceedings of the 2nd workshop on neural machine translation and generation, 36–44
21. Thompson B, Gwinnup J, Khayrallah H, Duh K, Koehn P (2019) Overcoming catastrophic forgetting during domain adaptation of neural machine translation. In: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, Vol 1 (Long and Short Papers), pp 2062–2068
22. Wang R, Utiyama M, Liu L, Chen K, Sumita E (2017) Instance weighting for neural machine translation domain adaptation. In: Proceedings of the 2017 conference on empirical methods in natural language processing, 1482–1488
23. Axelrod A, He X, Gao J (2011) Domain adaptation via pseudo in-domain data selection. In: Proceedings of the 2011 conference on empirical methods in natural language processing, 355–362
24. Chen B, Cherry C, Foster G, Larkin S (2017) Cost weighting for neural machine translation domain adaptation. In: Proceedings of the first workshop on neural machine translation, 40–46
25. Vilar D (2018) Learning hidden unit contribution for adapting neural machine translation models. In: Proceedings of the 2018 conference of the North American chapter of the association for computational linguistics: human language technologies, Vol 2 (Short Papers), pp 500–505
26. Yan S, Dahlmann L, Petrushkov P, Hewavitharana S, Khadivi S (2019) Word-based domain adaptation for neural machine translation. CoRR **abs/1906.03129**
27. Zhang X, Shapiro P, Kumar G, McNamee P, Carpuat M, Duh K (2019) Curriculum learning for domain adaptation in neural machine translation. In: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, Vol 1 (Long and Short Papers), pp 1903–1915
28. Zeng J, Liu, Y, Su J, Ge Y, Lu Y, Yin Y, Luo J (2019) Iterative dual domain adaptation for neural machine translation. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing, 845–855
29. Wang Z, Mehta S.V, Poczos B, Carbonell J.G (2020) Efficient meta lifelong-learning with limited memory. In: Proceedings of the 2020 conference on empirical methods in natural language processing, 535–548
30. Gulcehre C, Firat O, Xu K, Cho K, Barrault L, Lin H.-C, Bougares F, Schwenk H, Bengio Y (2015) On using monolingual corpora in neural machine translation. CoRR **abs/1503.03535**
31. Dou Z.-Y, Wang X, Hu J, Neubig G (2019) Domain differential adaptation for neural machine translation. In: Proceedings of the 3rd workshop on neural generation and translation, 59–69
32. Nguyen T.Q, Chiang D (2018) Improving lexical choice in neural machine translation. In: Proceedings of the 2018 conference of the North American chapter of the association for computational linguistics: human language technologies, Vol 1 (Long Papers), pp 334–343
33. Britz D, Le Q, Pryzant R (2017) Effective domain mixing for neural machine translation. In: Proceedings of the second conference on machine translation, 118–126
34. Wang Y, Wang L, Shi S, Li VO, Tu Z (2020) Go from the general to the particular: multi-domain translation with domain transformation networks. Proc AAAI Conf Artif Intell 34:9233–9241
35. Kobus C, Crego J.M, Senellart J (2017) Domain control for neural machine translation. In: Proceedings of the international conference recent advances in natural language processing, RANLP 2017, 372–378
36. Stergiadis E, Kumar S, Kovalev F, Levin P (2021) Multi-domain adaptation in neural machine translation through multidimensional tagging. CoRR **abs/2102.10160**
37. Thompson B, Khayrallah H, Anastasopoulos A, McCarthy A.D, Duh K, Marvin R, McNamee P, Gwinnup J, Anderson T, Koehn P (2018) Freezing subnetworks to analyze domain adaptation in neural machine translation. In: EMNLP 2018 third conference on machine translation (WMT18), pp 124–132. Association for Computational Linguistics
38. Wuebker J, Simianer P, DeNero J (2018) Compact personalized models for neural machine translation. In: Proceedings of the 2018 conference on empirical methods in natural language processing, 881–886
39. Gu S, Feng Y, Liu Q (2019) Improving domain adaptation translation with domain invariant and specific information. In: Proceedings of NAACL-HLT, 3081–3091
40. Wu S, Zhang D, Zhou M (2019) Effective soft-adaptation for neural machine translation. In: CCF International conference on natural language processing and Chinese computing, Springer, pp 254–264
41. Zeng J, Su J, Wen H, Liu Y, Xie J, Yin Y, Zhao J (2018) Multi-domain neural machine translation with word-level domain context discrimination. In: Proceedings of the 2018 conference on empirical methods in natural language processing, 447–457
42. Su J, Zeng J, Xie J, Wen H, Yin Y, Liu Y (2019) Exploring discriminative word-level domain contexts for multi-domain neural machine translation. IEEE Trans Pattern Anal Mach Intell 43(5):1530–1545
43. Jiang H, Liang C, Wang C, Zhao T (2020) Multi-domain neural machine translation with word-level adaptive layer-wise domain mixing. In: Proceedings of the 58th annual meeting of the association for computational linguistics, 1823–1834
44. Pham M.Q, Crego J.M, Yvon F, Senellart J (2020) A study of residual adapters for multi-domain neural machine translation. In:

Proceedings of the fifth conference on machine translation, 617–628
45. Sharaf A, Hassan H, Daumé III H (2020) Meta-learning for few-shot nmt adaptation. In: Proceedings of the Fourth workshop on neural generation and translation, 43–53
46. Zoph B, Yuret D, May J, Knight K (2016) Transfer learning for low-resource neural machine translation. In: Proceedings of the 2016 conference on empirical methods in natural language processing, pp 1568–1575
47. Sun T, Shao Y, Li X, Liu P, Yan H, Qiu X, Huang X (2020) Learning sparse sharing architectures for multiple tasks. Proc AAAI Conf Artif Intell 34:8936–8943
48. Lin Z, Wu L, Wang M, Li L (2021) Learning language specific sub-network for multilingual machine translation. In: Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing (Vol 1: Long Papers), p 293–305
49. Tian L, Wong D.F, Chao L.S, Quaresma P, Oliveira F, Yi L (2014) Um-corpus: a large English-Chinese parallel corpus for statistical machine translation. In: LREC, 1837–1842
50. Sennrich R, Haddow B, Birch A (2016) Neural machine translation of rare words with subword units. In: Proceedings of the 54th annual meeting of the association for computational linguistics (Vol 1: Long Papers), p 1715–1725
51. Papineni K, Roukos S, Ward T, Zhu W.-J (2002) Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th annual meeting of the association for computational linguistics, 311–318
52. Ott M, Edunov S, Baevski A, Fan A, Gross S, Ng N, Grangier D, Auli M (2019) fairseq: A fast, extensible toolkit for sequence modeling. In: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics (Demonstrations), 48–53
53. Kingma D.P, Ba J (2015) Adam: A method for stochastic optimization. In: ICLR (Poster)