Multi-Domain Neural Machine Translation with Word-Level Domain Context Discrimination

Jiali Zeng[1] **Jinsong Su**[1*] **Huating Wen**[1] **Yang Liu**[2] **Jun Xie**[3]
Yongjing Yin[1] **Jianqiang Zhao**[4]
[1]Xiamen University, Xiamen, China [2]Tsinghua University, Beijing, China
[3]Mobile Internet Group, Tencent Technology Co., Ltd, Beijing, China
[4]Meiya Pico information Co.,Ltd, Xiamen, China
lemon@stu.xmu.edu.cn, {jssu,htwen,yinyongjing}@xmu.edu.cn
liuyang2011@tsinghua.edu.cn, stiffxie@tencent.com, zhaojq@300188.cn

Abstract

With great practical value, the study of Multi-domain Neural Machine Translation (NMT) mainly focuses on using mixed-domain parallel sentences to construct a unified model that allows translation to switch between different domains. Intuitively, words in a sentence are related to its domain to varying degrees, so that they will exert disparate impacts on the multi-domain NMT modeling. Based on this intuition, in this paper, we devote to distinguishing and exploiting word-level domain contexts for multi-domain NMT. To this end, we jointly model NMT with monolingual attention-based domain classification tasks and improve NMT as follows: 1) Based on the sentence representations produced by a domain classifier and an adversarial domain classifier, we generate two gating vectors and use them to construct domain-specific and domain-shared annotations, for later translation predictions via different attention models; 2) We utilize the attention weights derived from target-side domain classifier to adjust the weights of target words in the training objective, enabling domain-related words to have greater impacts during model training. Experimental results on Chinese-English and English-French multi-domain translation tasks demonstrate the effectiveness of the proposed model. Source codes of this paper are available on Github https://github.com/DeepLearnXMU/WDCNMT.

1 Introduction

In recent years, neural machine translation (NMT) has achieved great advancement (Nal and Phil, 2013; Sutskever et al., 2014; Bahdanau et al., 2015). However, two difficulties are encountered in the practical applications of NMT. On the one hand, training a NMT model for a specific domain requires a large quantity of parallel sentences in such domain, which is often not readily available. Hence, the much more common practice is to construct NMT models using mixed-domain parallel sentences. In this way, the domain-shared translation knowledge can be fully exploited. On the other hand, the translated sentences often belong to multiple domains, thus requiring a NMT model general to different domains. Since the textual styles, sentence structures and terminologies in different domains are often remarkably distinctive, whether such domain-specific translation knowledge is effectively preserved could have a direct effect on the performance of the NMT model. Therefore, how to simultaneously exploit the exclusive and shared translation knowledge of mixed-domain parallel sentences for multi-domain NMT remains a challenging task.

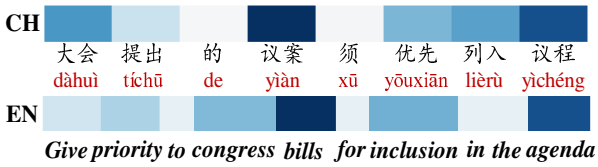


Figure 1: Word-level correlation heat map to *Laws* domain for a Chinese(CH)-English(EN) parallel sentence.

To tackle this problem, recently, researchers have carried out many constructive and in-depth studies (Kobus et al., 2016; Zhang et al., 2016; Pryzant et al., 2017; Farajian et al., 2017). However, most of these studies mainly focus on the utilization of domain contexts as a whole in NMT, while ignoring the discrimination of domain contexts at finer-grained level. In each sentence, some words are closely associated with its domain, while others are domain-independent. Intuitively, these two kinds of words play differ-

*Corresponding author

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 447–457
Brussels, Belgium, October 31 - November 4, 2018. ©2018 Association for Computational Linguistics

ent roles in multi-domain NMT, nevertheless, they are not being distinguished by the current models. Take the sentence shown in Figure 1 for example. The Chinese words "'大会"(congress), "议案"(bills), "列入"(inclusion), and "议程"(agenda) are frequently used in *Laws* domain and imply the *Laws* style of the sentence, while other words in this sentence are common in all domains and they mainly indicate the semantic meaning of the sentence. Thus, it is reasonable to distinguish and encode these two types of words separately to capture domain-specific and domain-shared contexts, allowing the exclusive and shared knowledge to be exploited without any interference from the other. Meanwhile, the English words "*priority*","*government*", "*bill*" and "*agenda*" are also closely related to *Laws* domain. To preserve the domain-related text style and idioms in generated translations, it is also reasonable for our model to pay more attention to these domain-related words than the others during model training. On this account, we believe that it is significant to distinguish and explore word-level domain contexts for multi-domain NMT.

In this paper, we propose a multi-domain NMT model with word-level domain context discrimination. Specifically, we first jointly model NMT with monolingual attention-based domain classification tasks. In source-side domain classification and adversarial domain classification tasks, we perform two individual attention operations on source-side annotations to generate the domain-specific and domain-shared vector representations of source sentence, respectively. Meanwhile, an attention operation is also placed on target-side hidden states to implement target-side domain classification. Then, we improve NMT with the following two approaches:

(1) According to the sentence representations produced by source-side domain classifier and adverisal domain classifier, we generate two gating vectors for each source annotation. With these two gating vectors, the encoded information of source annotation is selected automatically to construct domain-specific and domain-shared annotations, both of which are used to guide translation predictions via two attention mechanisms;

(2) Based on the attention weights of the target words from target-side domain classifier, we employ word-level cost weighting strategy to refine our model training. In this way, domain-specific target words will be assigned greater weights than others in the objective function of our model.

Our work demonstrates the benefits of separate modeling of the domain-specific and domain-shared contexts, which echoes with the successful applications of the multi-task learning based on shared-private architecture in many tasks, such as discourse relation recognition (Liu et al., 2017b), word segmentation (Chen et al., 2017b), text classification (Liu et al., 2017a), and image classification (Liu et al., 2016). Overall, the main contributions of our work are summarized as follows:

- We propose to construct domain-specific and domain-shared source annotations from initial annotations, of which effects are respectively captured for translation predictions.
- We propose to adjust the weights of target words in the training objective of NMT according to their relevance to different domains.
- We conduct experiments on large-scale multi-domain Chinese-English and English-French datasets. Experimental results demonstrate the effectiveness of our model.

2 Model

Figure 2 illustrates the architecture of our model, which includes a neural encoder equipped with a domain classifier and an adversarial domain classifier, and a neural decoder with two attention models and a target-side domain classifier.

2.1 Neural Encoder

As shown in the lower part of Figure 2, our encoder leverages the sentence representations produced by these two classifiers to construct domain-specific and domain-shared annotations from initial ones, preventing the exclusive and shared translation knowledge from interfering with each other. In our encoder, the input sentence $\mathbf{x}=x_1, x_2, ..., x_N$ are first mapped to word vectors and then fed into a bidirectional GRU (Cho et al., 2014) to obtain $\overrightarrow{\mathbf{h}}=\overrightarrow{h}_1, \overrightarrow{h}_2, ..., \overrightarrow{h}_N$ and $\overleftarrow{\mathbf{h}}=\overleftarrow{h}_1, \overleftarrow{h}_2, ..., \overleftarrow{h}_N$ in the left-to-right and right-to-left directions, respectively. These two sequences are then concatenated as $h_i = \{\overrightarrow{h}_i^\top, \overleftarrow{h}_i^\top\}^\top$ to form the word-level semantic representation of the input sentence.

Domain Classifier and Adversarial Domain Classifier. With annotations $\{h_i\}_{i=1}^N$, we employ

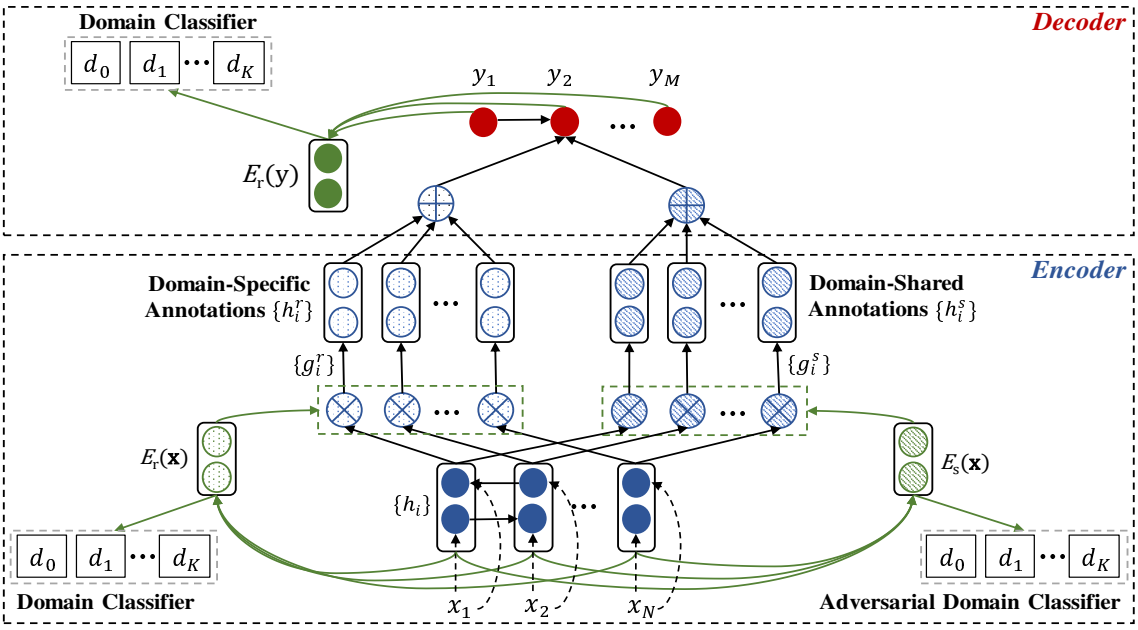


Figure 2: The architecture illustration of our model. Note that our two source-side domain classifiers are used to produce domain-specific and domain-shared annotations, respectively, and our target-side domain classifier is only used during model training.

two attention-like aggregators to generate the semantic representations of sentence $\mathbf{x}$, denoted by the vectors $E_r(\mathbf{x})$ and $E_s(\mathbf{x})$, respectively. Based on these two vectors, we employ the same neural network to model two classifiers with different context modeling objectives:

One is a domain classifier that aims to distinguish different domains in order to generate domain-specific source-side contexts. It is trained using the objective function $\mathcal{J}_{dc}^{s}(\mathbf{x};\boldsymbol{\theta}_{\boldsymbol{dc}}^{\boldsymbol{s}}) = \log p(d|\mathbf{x};\boldsymbol{\theta}_{\boldsymbol{dc}}^{\boldsymbol{s}})$, where d is the domain tag of $\mathbf{x}$ and $\boldsymbol{\theta}_{\boldsymbol{dc}}^{\boldsymbol{s}}$ is its parameter set. The other is an adversarial domain classifier capturing source-side domain-shared contexts. To this end, we train it using the following adversarial loss functions:

$$\mathcal{J}_{adc}^{s1}(\mathbf{x};\boldsymbol{\theta}_{\boldsymbol{adc}}^{\boldsymbol{s1}}) = \log p(d|\mathbf{x};\boldsymbol{\theta}_{\boldsymbol{adc}}^{\boldsymbol{s1}},\boldsymbol{\theta}_{\boldsymbol{adc}}^{\boldsymbol{s2}}), \quad (1)$$

$$\mathcal{J}_{adc}^{s2}(\mathbf{x};\boldsymbol{\theta}_{\boldsymbol{adc}}^{\boldsymbol{s2}}) = H(p(d|\mathbf{x};\boldsymbol{\theta}_{\boldsymbol{adc}}^{\boldsymbol{s1}},\boldsymbol{\theta}_{\boldsymbol{adc}}^{\boldsymbol{s2}})), \quad (2)$$

where $H(p(\cdot))=-\sum_{k=1}^{K} p_k(\cdot)\log\ p_k(\cdot)$ is an entropy of distribution $p(\cdot)$ with K domain labels, $\boldsymbol{\theta}_{\boldsymbol{adc}}^{\boldsymbol{s1}}$ and $\boldsymbol{\theta}_{\boldsymbol{adc}}^{\boldsymbol{s2}}$ denote the parameters of softmax layer and the generation layer of $E_s(\mathbf{x})$ in this classifier, respectively. By this means, $E_r(\mathbf{x})$ and $E_s(\mathbf{x})$ are expected to encode the domain-specific and domain-shared semantic representations of $\mathbf{x}$, respectively. It should be noted that our utilization of domain classifiers is similar to adversarial training used in (Pryzant et al., 2017) which injects domain-shared contexts into annotations. However, by contrast, we introduce domain classifier and adversarial domain classifier simultaneously to distinguish different kinds of contexts for NMT more explicitly.

Here we describe only the modeling procedure of the domain classifier, while it is also applicable to the adversarial domain classifier. Specifically, $E_r(\mathbf{x})$ is defined as follows:

$$E_r(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i h_i, \quad (3)$$

$$\text{where} \quad \alpha_i = \frac{exp(e_i)}{\sum_{i'}^{N} exp(e_{i'})},$$

$$e_i = (v_a)^{\top} \tanh(W_a h_i),$$

and v_a and W_a are the relevant attention parameters. Then, we feed $E_r(\mathbf{x})$ into a fully connected layer with *ReLU* function (Ballesteros et al., 2015), and then pass its output through a *softmax* layer to implement domain classification

$$p(\cdot|\mathbf{x};\boldsymbol{\theta}_{\boldsymbol{dc}}^{\boldsymbol{s}})$$

$$= softmax(W_{dc}^{s\top}\ ReLU(E_r(\mathbf{x})) + b_{dc}^{s}), \quad (4)$$

where W_{dc}^{s} and b_{dc}^{s} are softmax parameters.

Domain-Specific and Domain-Shared Annotations. Since domain-specific and domain-shared contexts have different effects on NMT, and thus

should be distinguished and separately captured by NMT model. Specifically, we first leverage the sentence representations $E_r(\mathbf{x})$ and $E_s(\mathbf{x})$ to generate two gating vectors, g_i^r and g_i^s, for annotation h_i in the following way:

$$g_i^r = sigmoid(W_{gr}^{(1)} E_r(\mathbf{x}) + W_{gr}^{(2)} h_i + b_{gr}), \tag{5}$$

$$g_i^s = sigmoid(W_{gs}^{(1)} E_s(\mathbf{x}) + W_{gs}^{(2)} h_i + b_{gs}), \tag{6}$$

where W_{gr}^*, W_{gs}^*, b_{gr} and b_{gs} denote the relevant matrices and bias, respectively. With these two vectors, we construct domain-specific and domain-shared annotations h_i^r and h_i^s from h_i:

$$h_i^r = g_i^r \odot h_i, \tag{7}$$

$$h_i^s = g_i^s \odot h_i. \tag{8}$$

2.2 Neural Decoder

The upper half of Figure 2 illustrates the architecture of our decoder. In particular, with the attention weights of target words from the domain classifier, we employ word-level cost weighting strategy to refine model training.

Formally, our decoder applies a nonlinear function $g(*)$ to define the conditional probability of translation $\mathbf{y}$=$y_1, y_2, ..., y_M$:

$$p(\mathbf{y}|\mathbf{x}) = \prod_{j=1}^{M} p(y_j|\mathbf{x}, y_{<j}) = \prod_{j=1}^{M} g(y_{j-1}, s_j, c_j^r, c_j^s), \tag{9}$$

where the vector s_j denotes the GRU hidden state. It is updated as

$$s_j = GRU(s_{j-1}, y_{j-1}, c_j^r, c_j^s). \tag{10}$$

Here the vectors c_j^r and c_j^s represent the domain-specific and domain-shared contexts, respectively.

Domain-Specific and Domain-Shared Context Vectors. When generating y_j, we define c_j^r as a weighted sum of the domain-specific annotations $\{h_i^r\}$:

$$c_j^r = \sum_{i=1}^{N} \frac{\exp(e_{j,i}^r)}{\sum_{i'=1}^{N} \exp(e_{j,i'}^r)} \cdot h_i^r, \tag{11}$$
$$\text{where} \quad e_{j,i}^r = a(s_{j-1}, h_i^r),$$

and a(*) is a feedforward neural network. Meanwhile, we produce c_j^s from the domain-shared annotations $\{h_i^s\}$ as in Eq. 11. By introducing c_j^r and c_j^s into s_j, our decoder is able to distinguish and simultaneously exploit two types of contexts for translation predictions.

Domain Classifier. We equip our decoder with a domain classifier with parameters $\boldsymbol{\theta_{tdc}}$, which maximizes the training objective i.e., $\mathcal{J}_{dc}^t(\mathbf{y}; \boldsymbol{\theta_{dc}^t}) = log\ p(d|\mathbf{y}; \boldsymbol{\theta_{dc}^t})$. To do this, we also apply attention operation to produce the domain-aware semantic representation $E_r(\mathbf{y})$ of $\mathbf{y}$,

$$E_r(\mathbf{y}) = \sum_{j=1}^{M} \beta_j s_j, \tag{12}$$
$$\text{where} \quad \beta_j = \frac{\exp(e_j)}{\sum_{j'}^{M} \exp(e_{j'})},$$
$$e_j = (v_b)^\top \tanh(W_b s_j),$$

and v_b and W_b are the related parameters. Likewise, we stack a domain classifier on top of $E_r(\mathbf{y})$. Note that this classifier is only used in model training to infer attention weights of target words. These weights measure their semantic relevance to different domains and can be utilized to adjust their cost weights in NMT training objective.

NMT Training Objective with Word-Level Cost Weighting. Formally, we define the objective function of NMT as follows:

$$\mathcal{J}_{nmt}(\mathbf{x}, \mathbf{y}; \boldsymbol{\theta_{nmt}}) = \sum_{j=1}^{M} (1 + \beta_j) \log p(y_j|\mathbf{x}, y_{<j}; \boldsymbol{\theta_{nmt}}), \tag{13}$$

where β_j is the attention weight of y_j obtained by Eq. (12), and $\boldsymbol{\theta_{nmt}}$ denotes the parameter set of NMT. By this scaling strategy, domain-specific words are emphasized with a bonus, while domain-shared words are updated as usual.

Please note that scaling costs with a multiplicative scalar essentially changes the magnitude of parameter update but without changing its direction (Chen et al., 2017a). Besides, although our scaling strategy is similar to the cost weighting proposed by Chen et al. (2017a), our approach differs from it in two aspects: First, we employ word-level cost weighting rather than sentence-level one to refine NMT training; Second, our approach is less time-consuming for multi-domain NMT.

2.3 Overall Training Objective

Given a mixed-domain training corpus $\mathcal{D} = \{(\mathbf{x}, \mathbf{y}, d)\}$, we train the proposed model accord-

ing to the following objective function:

$$\begin{aligned}\mathcal{J}(\mathcal{D};\boldsymbol{\theta}) = \sum_{(\mathbf{x},\mathbf{y},d)\in\mathcal{D}} \{&\mathcal{J}_{nmt}(\mathbf{x},\mathbf{y};\boldsymbol{\theta_{nmt}}) \\ &+ \mathcal{J}^{s}_{dc}(\mathbf{x};\boldsymbol{\theta^{s}_{dc}}) + \mathcal{J}^{t}_{dc}(\mathbf{y};\boldsymbol{\theta^{t}_{dc}}) \\ &+ \mathcal{J}^{s1}_{adc}(\mathbf{x};\boldsymbol{\theta^{s1}_{adc}}) + \lambda \cdot \mathcal{J}^{s2}_{adc}(\mathbf{x};\boldsymbol{\theta^{s2}_{adc}})\}\end{aligned} \quad (14)$$

where $\mathcal{J}_{nmt}(*)$, $\mathcal{J}^{s}_{dc}(*)$, $\mathcal{J}^{t}_{dc}(*)$ and $\mathcal{J}^{s*}_{adc}(*)$ are the objective functions of NMT, source-side domain classifier, target-side domain classifier, and source-side adversarial domain classifier, respectively, $\boldsymbol{\theta}$=$\{\boldsymbol{\theta_{nmt}}, \boldsymbol{\theta^{s}_{dc}}, \boldsymbol{\theta^{t}_{dc}}, \boldsymbol{\theta^{s1}_{adc}}, \boldsymbol{\theta^{s2}_{adc}}\}$, and λ is the hyper-parameter for adversarial learning.

Particularly, to ensure encoding accuracy of domain-shared contexts, we follow Chen et al. (2017b) to adopt an alternative two-phase strategy in training, where we alternatively optimize $\mathcal{J}(\mathcal{D};\boldsymbol{\theta})$ with $\boldsymbol{\theta^{s1}_{adc}}$ and $\{\boldsymbol{\theta}$-$\boldsymbol{\theta^{s1}_{adc}}\}$ respectively fixed at a time.

3 Experiment

To investigate the effectiveness of our model, we conducted multi-domain translation experiments on Chinese-English and English-French datasets.

3.1 Setup

Datasets. For Chinese-English translation, our data comes from UM-Corpus (Tian et al., 2014) and LDC[1]. To ensure data quality, we chose only the parallel sentences with domain label *Laws*, *Spoken*, and *Thesis* from UM-Corpus, and the LDC bilingual sentences related to *News* domain as our dataset. We used randomly selected sentences from UM-Corpus and LDC as development set, and combined the test set of UM-Corpus and randomly selected sentences from LDC to construct our test set. For English-French translation, we conducted experiments on the datasets of OPUS corpus[2], containing sentence pairs from *Medical*, *News*, and *Parliamentary* domains. We also divided these datasets into training, development and test sets. Table 1 provides the statistics of the corpora used in our experiments.

We performed word segmentation on Chinese sentences using *Stanford Segmenter*[3], and tokenized English and French sentences using *MOSES* script[4]. Then, we employed *Byte Pair Encoding* (Sennrich et al., 2016) to convert all words into subwords. The translation quality was evaluated by case-sensitive BLEU (Papineni et al., 2002).

Task	Domain	Train	Dev	Test
CH-EN	Laws	219K	600	456
	Spoken	219K	600	455
	Thesis	299K	800	625
	News	300K	800	650
EN-FR	Medical	1.09M	800	2000
	News	180K	800	2000
	Parliamentary	2.04M	800	2000

Table 1: Sentence numbers of data sets in our experiments.

Contrast Models. Since our model is essentially a standard attentional NMT model enhanced with **w**ord-level **d**omain **c**ontexts, we refer to it as **+WDC**. We compared it with the following models, namely:

- **OpenNMT**[5]. A famous open-source NMT system used widely in the NMT community trained on mix-domain training set.
- **DL4NMT-single** (Bahdanau et al., 2015). A reimplemented attentional NMT trained on a single domain dataset.
- **DL4NMT-mix** (Bahdanau et al., 2015). A reimplemented attentional NMT trained on mix-domain training set.
- **DL4NMT-finetune** (Luong and Manning, 2015). A reimplemented attentional NMT which is first trained using out-of-domain training corpus and then fine-tuned using in-domain dataset.
- **+Domain Control (+DC)** (Kobus et al., 2016). It directly introduces embeddings of source domain tag to enrich annotations of encoder.
- **+Multitask Learning (+ML1)** (Dong et al., 2015). It adopts a multi-task learning framework that shares encoder representation and separates the decoder modeling of different domains.
- **+Multitask Learning (+ML2)** (Pryzant et al., 2017). This model jointly trains

[1] https://www.ldc.upenn.edu/.
[2] http://opus.nlpl.eu/
[3] https://nlp.stanford.edu/
[4] http://www.statmt.org/moses/
[5] http://opennmt.net/.

NMT with domain classification via multi-task learning.

- **+Adversarial Discriminative Mixing (+ADM)** (Pryzant et al., 2017). It employs adversarial training to achieve the domain adaptation in NMT.

- **+Target Token Mixing (+TTM)** (Pryzant et al., 2017). This model is similar to +DC, with the only difference that it enriches source annotations by adding target-side domain tag rather than source-side one.

Note that our model uses two annotation sequences, thus we also compared it with the aforementioned models with two times of hidden state size ($2\times hd$). To further examine the effectiveness of the proposed components in our model, we also provided the performance of the following variants of our model:

- **+WDC(S)**. It only exploits the source-side word-level domain contexts for multi-domain NMT.

- **+WDC(T)**. It only employ word-level cost weighting on the target side to refine the model training.

Implementation Details. Following the common practice, we only used the training sentences within 50 words to efficiently train NMT models. Thus, 85.40% and 88.96% of the Chinese-English and English-French parallel sentences were covered in our experiments. In addition, we set the vocabulary size for Chinese-English and English-French as 32,000 and 32,000, respectively. In doing so, our vocabularies covered 99.97% Chinese words and 99.99% English words of the Chinese-English corpus, and almost 100% English words and 99.99% French words of the English-French corpus, respectively.

We applied *Adam* (Kingma and Ba, 2015) to train models and determined the best model parameters based on the model performance on development set. The used hyper-parameter were set as follows: β_1 and β_2 of Adam as 0.9 and 0.999, word embedding dimension as 500, hidden layer size as 1000, learning rate as 5×10^{-4}, batch size as 80, gradient norm as 1.0, dropout rate as 0.1, and beamsize as 10. Other settings were set following (Bahdanau et al., 2015).

Model	Laws	Spoken	Thesis	News
Contrast Models (**1**×**hd**)				
OpenNMT	45.82	9.15	13.93	19.73
DL4NMT-single	43.66	5.49	14.54	18.74
DL4NMT-mix	46.82	8.95	15.93	20.33
DL4NMT-finetune	54.19	8.77	16.71	21.55
+DC	49.83	9.18	16.71	20.58
+ML1	46.82	6.66	15.10	20.17
+ML2	48.95	9.45	15.85	20.48
+ADM	48.30	9.41	16.34	20.06
+TTM	49.05	9.36	16.42	20.44
Contrast Models (**2**×**hd**)				
DL4NMT-single	44.48	6.29	14.66	19.87
DL4NMT-mix	48.74	9.01	16.12	20.14
DL4NMT-finetune	54.69	9.07	17.11	21.85
+DC	50.43	9.38	16.45	20.44
+ML1	49.49	7.67	15.50	20.34
+ML2	50.05	9.35	16.03	20.64
+ADM	48.33	9.06	16.59	19.69
+TTM	49.92	9.01	16.38	21.04
Our Models				
+WDC(S)	54.55	10.12	17.22	22.16
+WDC(T)	51.94	9.76	17.72	21.02
+WDC	**55.03**	**10.20**	**18.04**	**22.29**

Table 2: Overall Evaluation of the Chinese-English translation task. $2\times hd$ = two times of hidden state size.

3.2 Results on Chinese-English Translation

We first determined the optimal hyper-parameter λ (see Eq. (14)) on the development set. To do this, we gradually varied λ from 0.1 to 1.0 with an increment of 0.1 in each step. Since our model achieved the best performance when λ=0.1, hence, we set λ=0.1 for all experiments thereafter.

Table 2 shows the overall experimental results. Using almost the same hyper-parameters, our re-implemented DL4NMT outperforms OpenNMT in all domains, demonstrating that our baseline is competitive in performance. Moreover, on all test sets of different domains, our model significantly outperforms other contrast models no matter which hyper-parameters they use. Furthermore, we arrive at the following conclusions:

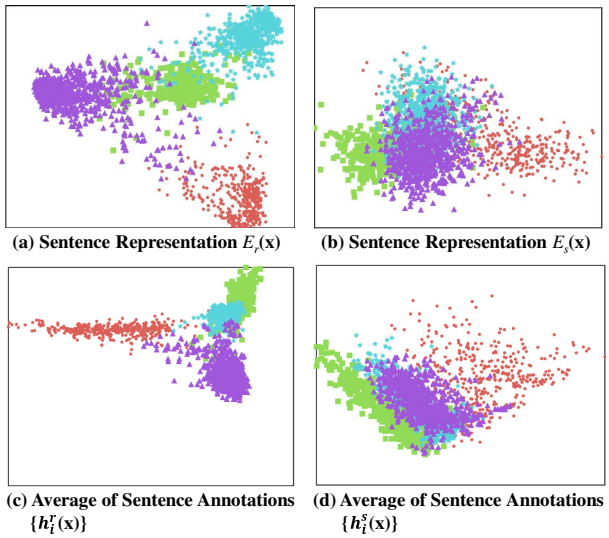
First, our model surpasses DL4NMT-single, DL4NMT-mix and DL4NMT-finetune, all of which are commonly used in domain adaptation for NMT. Please note that DL4NMT-finetune requires multiple adapted NMT models to be constructed, while ours is a unified one that works well in all domains.

Second, compared with +DC, +ML2 and +ADM which all exploit source-side domain contexts for multi-domain NMT, our +WDC(S) still

(a) An Example Sentence in *Laws* Domain

(b) An Example Sentence in *Thesis* Domain

Figure 3: The correlation heat map of the gating vectors(blue/green) to domain-specific/domain-shared annotations in two example sentences. Note that domain-specific words "澳门"(Macao), "立法会"(Legislative Council), "产生"(Formation), "办法"(Method), "封闭"(Seal), "计算"(Calculation), "实验" (Experiment) are strengthened by g_i^r, while most of the domain-shared words "的"(of) and "与"(and) are focused by g_i^s.



(a) Sentence Representation $E_r(\mathbf{x})$ (b) Sentence Representation $E_s(\mathbf{x})$

(c) Average of Sentence Annotations $\{h_i^r(\mathbf{x})\}$ (d) Average of Sentence Annotations $\{h_i^s(\mathbf{x})\}$

Figure 4: The visualization of the sentence representations and their corresponding average annotations, where the triangle-shaped(purple), circle-shaped(red), square-shaped(green) and pentagonal-shaped(blue) points denote *News*, *Laws*, *Spoken* and *Thesis* sentences, respectively.

exhibits better performance. This is because that these models focus on one aspect of domain contexts, while our model considers both domain-specific and domain-shared contexts on the source side.

Third, +WDC(T) also outperforms DL4NMT, revealing that it is reasonable and effective to emphasize domain-specific words in model training..

Last, +WDC achieves the best performance when compared with both +WDC(S) and +WDC(T). Therefore, we believe that word-level domain contexts on the both sides are complementary to each other, and utilizing them simultaneously is beneficial to multi-domain NMT.

3.3 Experimental Analysis

Furthermore, we conducted several visualization experiments to empirically analyze the individual effectiveness of the added model components.

3.3.1 Visualizations of Gating Vectors

We first visualized the gating vectors g_i^r and g_i^s to quantify their effects on extracting domain-specific and domain-shared contexts from initial source-side annotations. Since both g_i^r and g_i^s are high dimension vectors, which are difficult to be visualized directly, we followed Li et al. (2016) and Zhou et al. (2017) to visualize their individual contributions to the final output, which can be approximated by their first derivatives.

Figure 3 shows the first derivative heat maps for two example sentences in *Laws* and *Thesis* domain, respectively. We can observe that without any loss of semantic meanings from source sentences, most of the domain-specific words are strengthened by g_i^r, while most of the domain-shared words, especially function words, are focused by g_i^s. This result is consistent with our expectation for the function of two gating vectors.

3.3.2 Visualizations of Sentence Representations and Annotations

Furthermore, we applied the *hypertools* (Heusser et al., 2018) to visualize the sentence representations $E_r(\mathbf{x})$ and $E_s(\mathbf{x})$, and the domain-specific and domain-shared annotation sequences $\{h_i^r\}_{i=1}^N$ and $\{h_i^s\}_{i=1}^N$. Here we represent each annotation sequence with its average vector in the figure.

As shown in Figure 4 (a) and (b), the sentence representation vectors and the average annotation vectors of different domains are clearly distributed in different regions. By contrast, their distributions are much more concentrated in Figure 4 (c) and (d). Thus, we conclude that our model is able to distinctively learn domain-specific and domain-shared contexts. Moreover, from Figure 4 (b), we observe that the sentence representation vectors of *Laws* domain does not completely coincide with

Domain	Top10 Target Words
Laws	*Article*, *Chapter*, *Principles*, *regulations*, *Provisions*, *Political*, *Servants*, *specify*, *China*, Municipal
Spoken	*meanly*, *Rusty*, *1910s*, *scours*, *mountaintops*, *paralyze*, *Puff*, *perpetrators*, *hitter*, *weightlifting*
Thesis	*aggregation*, *Activities*, *Computation*, *Alzheimer*, *nn*, *Contemporarily*, *EVALUATION*, *ethoxycarbonyl*, *sCRC*, *Announced*
News	*months*, *agency*, *outweighed*, *unconstitutionally*, *Congolese*, *session*, *Asia*, *news*, *hurts*, *francs*

Table 3: Examples of Domain-Specific Target Words.

those of the other domains, this may be caused by the more formal and consistent sentence styles in *Laws* domain.

3.3.3 Illustrations of Domain-Specific Target Words

Lastly, for each domain, we presented the top ten target words with the highest weights learned by our target-side domain classifier. To do this, we calculated the average attention weight of each word in the training corpus as its corresponding domain weight.

As is clearly shown in Table 3 that most listed target words are closely related to their domains. This result validates the aforementioned hypothesis that some words are domain-dependent while others are domain-independent, and our target-side domain classifier is capable of distinguishing them with different attention weights.

3.4 Results on English-French Translation

Likewise, we determined the optimal λ=0.1 on the development set. Table 4 gives the results of English-French multi-domain translation. Similar to the previous experimental result in Section 3.2, our model continues to achieve the best performance compared to all contrast models using two different hidden state size settings, which demonstrates again that our model is effective and general to different language pairs in multi-domain NMT.

4 Related Work

In this work, we study on multi-domain machine translation in the field of domain adaptation for machine translation, which has attracted great attention since SMT (Clark et al., 2012; Huck et al., 2015; Sennrich et al., 2013). As for NMT, the dominant strategies for domain adaptation generally fall into two categories:

Model	Medical	Parliamentary	News
Contrast Models (**1**×**hd**)			
OpenNMT	78.78	32.96	30.22
DL4NMT-single	77.34	33.28	29.56
DL4NMT-mix	78.48	33.16	31.62
DL4NMT-finetune	78.61	33.72	34.04
+DC	79.34	33.38	33.94
+ML1	77.29	33.39	31.92
+ML2	78.65	33.55	33.48
+ADM	76.74	33.06	33.43
+TTM	78.27	33.29	33.37
Contrast Models (**2**×**hd**)			
DL4NMT-single	78.50	33.38	30.23
DL4NMT-mix	78.84	33.19	33.28
DL4NMT-finetune	79.17	33.88	34.20
+DC	79.96	33.44	33.52
+ML1	78.38	33.20	31.90
+ML2	79.41	33.55	33.62
+ADM	79.31	33.50	33.34
+TTM	79.36	33.13	33.68
Our Models			
+WDC(S)	82.76	34.13	34.31
+WDC(T)	81.51	33.76	33.78
+WDC	**83.35**	**34.17**	**34.87**

Table 4: Overall Evaluation on the English-French translation task.

The first category is to transfer out-of-domain knowledge to in-domain translation. The conventional method is *fine-tuning*, which first trains the model on out-of-domain dataset and then fine-tunes it on in-domain dataset (Luong and Manning, 2015; Zoph et al., 2016; Servan et al., 2016). Freitag and Al-Onaizan (2016) proceeded further by ensembling the fine-tuned model with the original one. Chu et al. (2017) fine-tuned the model using the mix of in-domain and out-of-domain training corpora. From the perspective of data selection, Chen et al. (2017a) scaled the top-level costs of NMT system according to each training sentence's similarity to the development set. Wang et al. (2017a) explored the data selection strategy based on sentence embeddings for NMT domain adaptation. Moreover, Wang et al. (2017b) further proposed several sentence and domain weighting methods with a dynamic weight learning strategy. However, these approaches usually only perform well on target domain while being highly time consuming in transferring translation knowledge to all the constitute domains.

The second category is to directly use a mixed-

domain training corpus to construct NMT model for the translated sentences derived from different domains. In this aspect, Kobus et al. (2016) incorporated domain information into NMT by appending a domain indicator token to each source sequence. Similarly, Johnson et al. (2016) added an artificial token to the input sequence to indicate the required target language. Contrastingly, Farajian et al. (2017) utilized the similarity between each test sentence and the training instances to dynamically set the hyper-parameters of the learning algorithm and update the generic model on the fly. Pryzant et al. (2017) proposed three novel models: *discriminative mixing* that jointly models NMT with domain classification, *adversarial discriminative mixing*, and *target token mixing* which appends a domain token to the target sequence. Sajjad et al. (2017) explored data concatenation, model stacking, data selection and multi-model ensemble to train multi-domain NMT. By exploiting domain as a tag or a feature, Tars and Fishel (2018) treated text domains as distinct languages in order to use multi-lingual approaches when implementing multi-domain NMT. Inspired by topic-based SMT, some researchers resorted to incorporating topical contexts into NMT. Chen et al. (2016) used the topic information of input sentence as an additional input to decoder. Zhang et al. (2016) enhanced the word representation by adding its topic embedding. However, these methods require to have explicit document boundaries between training data, which unfortunately do not exist in most datasets.

Overall, our work is related to the second type of approach with (Pryzant et al., 2017) and (Chen et al., 2017a) most related to ours. Unlike (Pryzant et al., 2017) applying adversarial training to only capture domain-shared translation knowledge, we further exploit domain-specific translation knowledge for multi-domain NMT. Also, in sharp contrast to (Chen et al., 2017a), our model not only exploits the source-side word-level domain contexts differently, but also employs a word-level cost weighting strategy for multi-domain NMT.

5 Conclusion and Future Work

In this work, we have explored how to utilize word-level domain contexts for multi-domain NMT. By jointly modeling NMT and domain classification tasks, we utilize the sentence representations of source-side domain classifier and adversarial domain classifier to construct domain-specific and domain-shared source annotations, which are then exploited by decoder. Moreover, using the attentional weights of target-side domain classifier, we adjust the weights of target words in the training objective to refine model training. Experimental results and in-depth analyses demonstrate the effectiveness of the proposed model.

In the future, we would like to extend the proposed word-level cost weighting strategy to source words. Besides, our method is also general to other NMT models. Therefore, we plan to apply our method to the NMT with complex architectures, for example, lattice-to-sequence NMT (Su et al., 2017), hierarchy-to-sequence NMT (Su et al., 2018), NMT with context-aware encoder (Zhang et al., 2017) and Transformer (Vaswani et al., 2017) and so on.

Acknowledgments

The authors were supported by National Natural Science Foundation of China (No. 61672440), the Fundamental Research Funds for the Central Universities (Grant No. ZK1024), and Scientific Research Project of National Language Committee of China (Grant No. YB135-49). We also thank the reviewers for their insightful comments.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proc. of ICLR 2015*.

Miguel Ballesteros, Chris Dyer, and Noah A. Smith. 2015. Improved transition-based parsing by modeling characters instead of words with lstms. In *Proc. of EMNLP 2015*.

Boxing Chen, Colin Cherry, George Foster, and Samuel Larkin. 2017a. Cost weighting for neural machine translation domain adaptation. In *Proc. of the First Workshop on Neural Machine Translation*.

Wenhu Chen, Evgeny Matusov, Shahram Khadivi, and Jan-Thorsten Peter. 2016. Guided alignment training for topic-aware neural machine translation. *CoRR abs/1607.01628*.

Xinchi Chen, Zhan Shi, Xipeng Qiu, and Xuanjing Huang. 2017b. Adversarial multi-criteria learning for chinese word segmentation. In *Proc. of ACL 2017*.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger

Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proc. of EMNLP 2014*.

Chenhui Chu, Raj Dabre, and Sadao Kurohashi. 2017. An empirical comparison of domain adaptation methods for neural machine translation. In *Proc. of ACL 2017*.

Jonathan H. Clark, Alon Lavie, and Chris Dyer. 2012. One system, many domains: Open-domain statistical machine translation via feature augmentation. In *Proc. of AMTA 2012*.

Daxiang Dong, Hua Wu, Wei He, Dianhai Yu, and Haifeng Wang. 2015. Multi-task learning for multiple language translation. In *Proc. of ACL 2015*.

M. Amin Farajian, Marco Turchi, Matteo Negri, and Marcello Federico. 2017. Multi-domain neural machine translation through unsupervised adaptation. In *Proc. of WMT 2017*.

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

Andrew C. Heusser, Kirsten Ziman, Lucy L. W. Owen, and Jeremy R. Manning. 2018. Hypertools: a python toolbox for gaining geometric insights into high-dimensional data. *Journal of Machine Learning Research*.

M Huck, A Birch, and B Haddow. 2015. Mixed-domain vs. multi-domain statistical machine translation. In *Proc. of MT Summit 2015*.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda B. Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, , and Jeffrey Dean. 2016. Google's multilingual neural machine translation system: Enabling zero-shot translation. *CoRR abs/1611.04558*.

Diederik P. Kingma and Jimmy Lei Ba. 2015. Adam: A method for stochastic optimization. In *Proc. of ICLR 2015*.

Catherine Kobus, Josep Crego, and Jean Senellart. 2016. Domain control for neural machine translation. *CoRR abs/1612.06140*.

Jiwei Li, Xinlei Chen, Eduard Hovy, and Dan Jurafsky. 2016. Visualizing and understanding neural models in nlp. In *Proc. of NAACL 2016*.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2016. Domain separation networks. In *Proc. of NIPS 2016*.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2017a. Adversarial multi-task learning for text classification. In *Proc. of ACL 2017*.

Yang Liu, Sujian Li, Xiaodong Zhang, and Zhifang Sui. 2017b. Implicit discourse relation classification via multi-task neural networks. In *Proc. of ACL 2017*.

Minh-Thang Luong and Christopher D Manning. 2015. Stanford neural machine translation systems for spoken language domains. In *Proc. of IWSLT 2015*.

Kalchbrenner Nal and Blunsom Phil. 2013. Recurrent continuous translation models. In *Proc. of EMNLP 2013*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proc. of ACL 2002*.

Reid Pryzant, Denny Britz, and Q Le. 2017. Effective domain mixing for neural machine translation. In *Proc. of WMT 2017*.

Hassan Sajjad, Nadir Durrani, Fahim Dalvi, Yonatan Belinkov, and Stephan Vogel. 2017. Neural machine translation training in a multi-domain scenario. *CoRR abs/1708.08712*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proc. of ACL 2016*.

Rico Sennrich, Holger Schwenk, and Walid Aransa. 2013. A multi-domain translation model framework for statistical machine translation. In *Proc. of ACL 2013*.

Christophe Servan, Josep Crego, and Jean Senellart. 2016. Domain specialization: a post-training domain adaptation for neural machine translation. *CoRR abs/1612.06141*.

Jinsong Su, Zhixing Tan, Deyi Xiong, Rongrong Ji, Xiaodong Shi, and Yang Liu. 2017. Lattice-based recurrent neural network encoders for neural machine translation. In *Proc. of AAAI 2017*, pages 3302–3308.

Jinsong Su, Jiali Zeng, Deyi Xiong, Yang Liu, Mingxuan Wang, and Jun Xie. 2018. A hierarchy-to-sequence attentional neural machine translation model. *IEEE/ACM Trans. Audio, Speech & Language Processing*, 26(3):623–632.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Proc. of NIPS 2014*.

Sander Tars and Mark Fishel. 2018. Multi-domain neural machine translation. *CoRR abs/1805.02282*.

Liang Tian, Derek F. Wong, Lidia S. Chao, Paulo Quaresma, Francisco Oliveira, Shuo Li, Yiming Wang, and Yi Lu. 2014. Um-corpus: A large english-chinese parallel corpus for statistical machine translation. In *Proc. of LREC 2014*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proc. of NIPS 2017*.

Rui Wang, Andrew Finch, Masao Utiyama, and Eiichiro Sumita. 2017a. Sentence embedding for neural machine translation domain adaptation. In *Proc. of ACL 2017*.

Rui Wang, Masao Utiyama, Lemao Liu, Kehai Chen, and Eiichiro Sumita. 2017b. Instance weighting for neural machine translation domain adaptation. In *Proc. of EMNLP 2017*.

Biao Zhang, Deyi Xiong, Jinsong Su, and Hong Duan. 2017. A context-aware recurrent encoder for neural machine translation. *IEEE/ACM Trans. Audio, Speech & Language Processing*, 25(12):2424–2432.

Jian Zhang, Liangyou Li, Andy Way, and Qun Liu. 2016. Topic-informed neural machine translation. In *Proc. of COLING 2016*.

Qingyu Zhou, Nan Yang, Furu Wei, and Ming Zhou. 2017. Selective encoding for abstractive sentence summarization. In *Proc. of ACL 2017*.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. *Proc. of EMNLP 2016*.